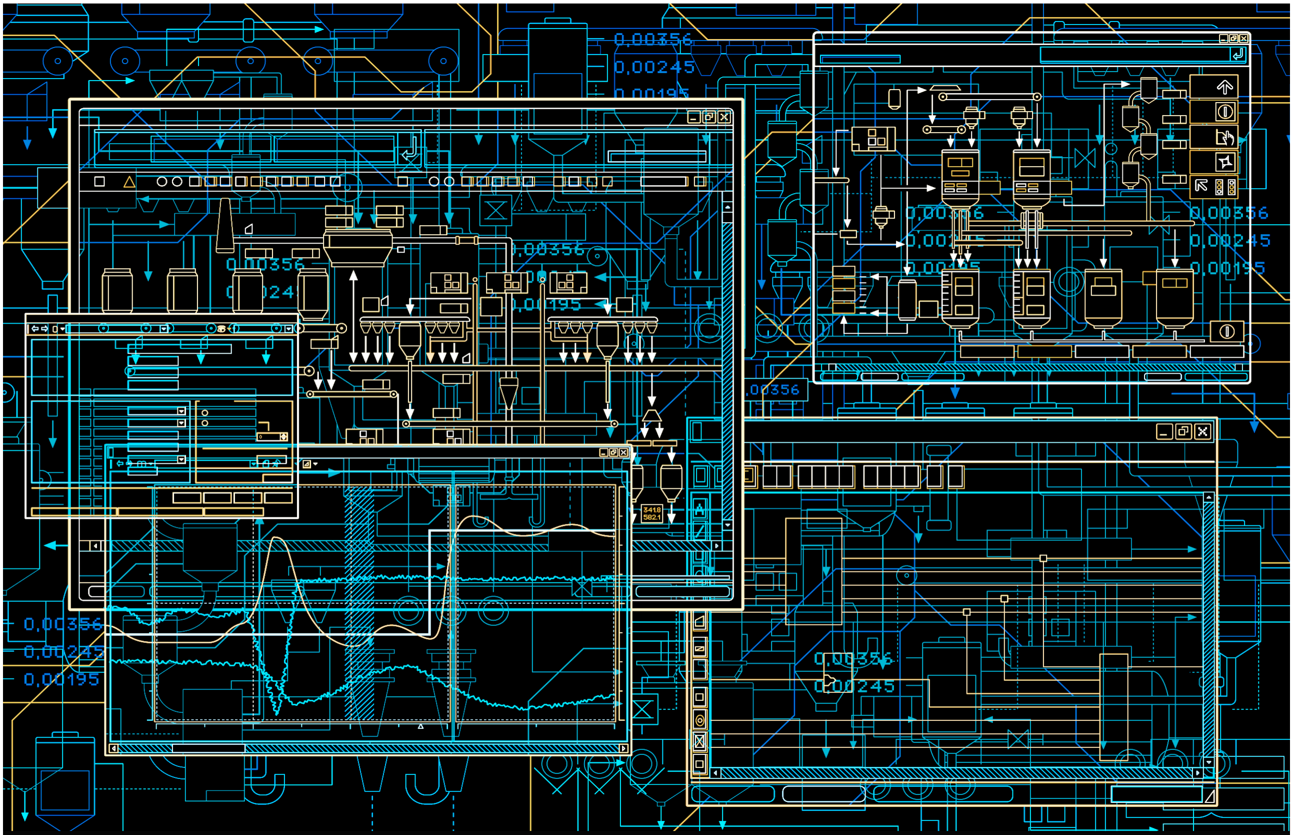


Device Management

PROFIBUS DTM Builder 6.0

System Version 6.0

Power and productivity
for a better world™

Device Management

PROFIBUS DTM Builder 6.0

System Version 6.0

NOTICE

This document contains information about one or more ABB products and may include a description of or a reference to one or more standards that may be generally relevant to the ABB products. The presence of any such description of a standard or reference to a standard is not a representation that all of the ABB products referenced in this document support all of the features of the described or referenced standard. In order to determine the specific features supported by a particular ABB product, the reader should consult the product specifications for the particular ABB product.

ABB may have one or more patents or pending patent applications protecting the intellectual property in the ABB products described in this document.

The information in this document is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any errors that may appear in this document.

Products described or referenced in this document are designed to be connected, and to communicate information and data via a secure network. It is the sole responsibility of the system/product owner to provide and continuously ensure a secure connection between the product and the system network and/or any other networks that may be connected.

The system/product owners must establish and maintain appropriate measures, including, but not limited to, the installation of firewalls, application of authentication measures, encryption of data, installation of antivirus programs, and so on, to protect the system, its products and networks, against security breaches, unauthorized access, interference, intrusion, leakage, and/or theft of data or information.

ABB verifies the function of released products and updates. However system/product owners are ultimately responsible to ensure that any system update (including but not limited to code changes, configuration file changes, third-party software updates or patches, hardware change out, and so on) is compatible with the security measures implemented. The system/product owners must verify that the system and associated products function as expected in the environment they are deployed.

In no event shall ABB be liable for direct, indirect, special, incidental or consequential damages of any nature or kind arising from the use of this document, nor shall ABB be liable for incidental or consequential damages arising from use of any software or hardware described in this document.

This document and parts thereof must not be reproduced or copied without written permission from ABB, and the contents thereof must not be imparted to a third party nor used for any unauthorized purpose.

The software or hardware described in this document is furnished under a license and may be used, copied, or disclosed only in accordance with the terms of such license. This product meets the requirements specified in EMC Directive 2004/108/EC and in Low Voltage Directive 2006/95/EC.

TRADEMARKS

All rights to copyrights, registered trademarks, and trademarks reside with their respective owners.

Release: April 2016
Document number: 3BDD011945-600 A

Table of Contents

About This User Manual

Section 1 - Introduction

Section 2 - Getting Started

Section 3 - PROFIBUS DTM Builder

Index

About This User Manual

General

Any security measures described in this document, for example, for user access, password security, network security, firewalls, virus protection, etc., represent possible steps that a user of an 800xA System may want to consider based on a risk assessment for a particular application and installation. This risk assessment, as well as the proper implementation, configuration, installation, operation, administration, and maintenance of all relevant security related equipment, software, and procedures, are the responsibility of the user of the 800xA System.

This User Manual contains a detailed description of how to use the PROFIBUS DTM Builder. The PROFIBUS DTM Builder is an enhancement of the Basic PROFIBUS DTM, which is described in Basic PROFIBUS DTM, Configuration manual.

For the latest information, refer to the corresponding Release Notes.

Document Conventions

Microsoft Windows conventions are normally used for the standard presentation of material when entering text, key sequences, prompts, messages, menu items, screen elements, etc.

Warning, Caution, Information, and Tip Icons

This publication includes **Caution**, and **Information** where appropriate to point out safety-related or other important information. It also includes **Tips** to point out useful hints to the reader. The corresponding symbols should be interpreted as follows:

⚡ Electrical warning icon indicates the presence of a hazard which could result in *electrical shock.*

⚠ Warning icon indicates the presence of a hazard which could result in *personal injury.*

❗ The caution icon indicates important information or a warning related to the concept discussed in the text. It may indicate the presence of a hazard, which could result in *software corruption or damage to equipment/property.*

ℹ The information icon draws the reader's attention to pertinent facts and conditions.

💡 The tip icon provides advice about how, for example, to design your project or use a certain function.

Although **Warning** hazards are related to personal injury, and **Caution** hazards are associated with equipment or property damage, it should be understood that operation of damaged equipment could, under certain operational conditions, result in degraded process performance leading to personal injury or death. Therefore, **fully comply** with all **Warning** and **Caution** notices.

Terminology

A complete and comprehensive list of Terms is included in the *System 800xA, System Guide, Functional Description (3BSE038018*)*. The listing includes terms and definitions as they apply to the 800xA system where the usage is different from commonly accepted industry standard definitions and definitions given in standard dictionaries such as ***Webster's Dictionary of Computer Terms***. Terms that uniquely apply to this instruction may be included here as part of this document.

The following is a list of terms associated with the Basic PROFIBUS DTM/PROFIBUS DTM Builder that one should be familiar with. The list contains terms and abbreviations that are unique to ABB or have a usage or definition that is different from standard industry usage.

Term/Acronym	Description
Basic PROFIBUS DTM	DTM for PROFIBUS devices, additionally serving as a runtime environment for device-specific DTMs, that have been built with the PROFIBUS DTM Builder
Device Type Manager (DTM)	Software component (device driver) for configuring, diagnosing, forcing, displaying the measured variables, etc. of a field device. It is familiar with the way the device works and supplies device-specific documentation.
Device Description Language (DDL)	Interpretable language for the formal description of device parameters
Frame Application (FA)	Frame application (run time environment) in accordance with the FDT specification for operating DTMs
Fieldbus Builder (FBB)	ABB aspect system for fieldbus and DTM management, implementing a frame application according to specification FDT 1.2
Field Device Tool (FDT)	The FDT concept describes the interface between a frame application and the device-specific software (DTM = Device Type Manager) of the device manufacturer. It enables devices produced by different manufacturers and different fieldbuses to be integrated in a single system. Currently supporting fieldbus protocols for PROFIBUS and HART.
GUI	Graphical interface of the application for user interactions

Term/Acronym	**Description**
Process Portal A (PPA)	System 800xASystem Product containing functionality for efficient control and supervision of an automated process. Key functions are presentation of graphics, process dialogs and presentation of alarms and trends.Platform for the integration of different aspect systems, for example, Fieldbus Builder PROFIBUS/HART
PROFIBUS DTM Builder	Software product consisting of an DP V1 editor to define device specific PROFIBUS DTMs, to be executed by the Basic PROFIBUS DTM.
System Application	A software package that provides functionality in the System 800xA. System applications cooperate according to rules defined by the System 800xA architecture, using mechanism provided by the Process Portal A. They are normally bundled into System Products. To participate in Aspect Object operations, and thus be an integrated part of an System 800xA, a system application must present itself as an aspect system. When there is no risk for confusion with user application, the term application may be used instead of system application.

Related Documentation

A complete list of all documents applicable to the *System 800xA, Extended Automation System is provided in Released User Documents, (3BUA000263*)*. This document lists applicable Release Notes and User Instructions. It is provided in PDF format and is included on the Release Notes/Documentation media provided with your system. Released User Documents are updated with each release and a new file is provided that contains all user documents applicable for that release with their applicable document number. Whenever a reference to a specific instruction is made, the instruction number is included in the reference.

The table below contains a list of relevant documentation.

Category	Title	Description
DTM	Basic PROFIBUS DTM, Installation (3BDD011941*)	The document describes how to install Basic PROFIBUS DTM with PROFIBUS DTM Builder expansion.
	Device Management, PROFIBUS DTM, (3BDD011938*)	The document describes the basic features and operation of the DTM applications.
	Basic PROFIBUS DTM, Release Notes (3BDD 011943*)	This document provides a brief overview on functionality and enumerates known problems encountered in the final testing of this product release.

Section 1 Introduction

Product Overview

ABB provides a Basic DTM called the Basic PROFIBUS DTM for PROFIBUS field devices which do not have a dedicated DTM, with the expansion of the Basic PROFIBUS DTM to the PROFIBUS DTM Builder, dedicated DTMs can be developed using the Basic PROFIBUS DTM supported applications. The Basic PROFIBUS DTM/PROFIBUS DTM Builder enables the field devices to be operated in a frame application conforming to FDT 1.2.

The base functionality, supported application and limits are described in the *Basic PROFIBUS DTM - Configuration* manual and the dedicated Release Notes. The *Basic PROFIBUS DTM - Configuration* manual describes additional features belonging to the DTM Builder expansion.

A general workflow on how to create a simple device-specific Basic PROFIBUS DTM based on the only GSD file is also included in *Device Management, PROFIBUS DTM (3BDD011938*)* and will not be described in this document.

Product Scope

The PROFIBUS DTM Builder expansion includes additional DTM applications, which can be started if the valid DTM license is included in the Registration application of the DTM. These additional DTM applications are as follows:

- **Application Editor**
  The Application editor allows the user to create device specific templates, which can be used within the Basic PROFIBUS DTM.
- **Parameterization**
  Offers configuration function for a device type in online mode.

- **Observation**
  Shows the process values of the connected device type in online mode.

Installation

The installation is described in *Basic PROFIBUS DTM, Installation (3BDD011941*)*.

Registration

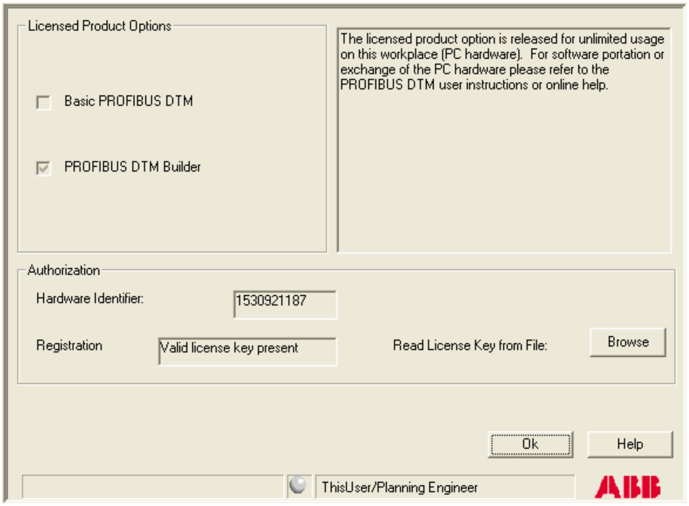
A separate license key is required for the PROFIBUS DTM Builder. This is associated with the one workstation computer on which the software is to be installed and operated.

To order and register the license key, follow the steps described below:

1. Install the DTM software on the workstation computer on which it will be used.
2. Create an instance of the Basic PROFIBUS DTM visible as *ABB GPB/Placeholder* in the FDT frame tool.
3. Open the DTM application “*Registration*”.
4. Copy the hardware identifier indicated for the workstation computer in a e-mail and send it with request description to DE-Automation Supportline de.automation.supportline@de.abb.com.

Notice that the PROFIBUS DTM Builder is for ABB internal use only. A license request will be surveyed, if the requirements are met.

It is strictly prohibited to give spread created template files based on the DTM Builder as well as license keys to external customers. Only a limited ABB internal support is offered.



Licensed Product Options

☐ Basic PROFIBUS DTM

☑ PROFIBUS DTM Builder

The licensed product option is released for unlimited usage on this workplace (PC hardware). For software portation or exchange of the PC hardware please refer to the PROFIBUS DTM user instructions or online help.

Authorization

Hardware Identifier: 1530921187

Registration: Valid license key present

Read License Key from File:

ThisUser/Planning Engineer

Figure 1. Registering the License Key

If a license key file has been shared, follow the procedure described below:

1. Open the DTM “*Registration”* application.
2. Click on the [**Browse**] button to import the license key. Once the file has been read in and the data verified as correct, the message "Valid license key present" appears. The DTM can be used on this workstation without any limitation. If the message "Invalid license key present" appears, check whether the license key is
   - for a different product
   - for a different workstation or
   - for a workstation with different hardware resources, for example, Ethernet interface card.

Intended User

This configuration guide is designed for application engineers and commissioning engineers. It explains how to build device-specific DTMs using the PROFIBUS DTM Builder. Those using this manual should be familiar with the basic method of operation of the PROFIBUS protocol.

Notes for use

The PROFIBUS DTM Builder supports all UNICODE characters, except the following, which shall not be used in any Graphical User Interface (GUI):

- &
- <
- >
- "
- '
- \t
- \r
- \n

REAL values use "." as decimal separator, independent from the language settings. The decimal separator “,” is not supported.

Section 2 Getting Started

Overview

This section describes how to use the PROFIBUS Application Editor. The Application Editor is a tool to create and draw up Graphical User Interfaces (GUI) with parameters to access device types through the Basic PROFIBUS DTM runtime environment.

This section describes the general workflow for creating a DTM application. The screens show both the existing and new controls.

Prerequisites and Requirements

The following requirements must be met, before carrying out the actions described later in this section:

- A FDT frame tool conforming to FDT 1.2 is installed and running.
- PROFIBUS Communication DTM is installed.
- Basic PROFIBUS DTM is installed as described in *Fieldbus, Basic PROFIBUS DTM, Installation (3BDD011938*)*.
- PROFIBUS DTM Builder license has been included in the Registration application, using the Placeholder DTM of the Basic PROFIBUS DTM.

Only if the DTM Builder license is included in the DTM Builder application, the Application Editor and DPV1 tab can be viewed. If the DTM Builder application is not visible in the DTM menu, an incorrect or missing DTM Builder license might be the reason. Check in the Registration application, whether the check box is selected for the DTM Builder box.

- A device-specific Basic PROFIBUS DTM and instance has been created as described in *Device Management, PROFIBUS DTM (3BDD011938*)*.

Utilizing the PROFIBUS DTM Builder

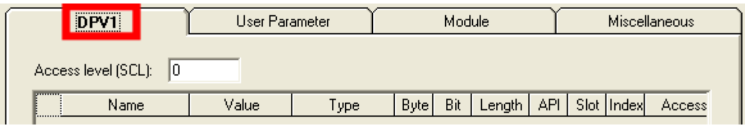
The following is a general workflow to create simple DTM application with DPV1 parameter. For more complex DTM applications, contact ABB Consulting for support and training.

In this section the creation of DTM application is described using the controls in the **Toolbox Area**. These controls are placed in the **Application Editor Area** by using the drag'n drop functionality provided by the Windows system. It is possible to use the Application Explorer Area as an alternative to the drag'n drop functionality of Tool Box Area. Right-click on the Application Explorer Area and select the required control. The control must be moved and placed manually in the **Application Editor Area** afterwards.

DPV1 Parameter Definition

To create device-specific applications, it is mandatory to define and include those DPV1 parameters, which are required to read or write data through acyclic communication to or from the device.

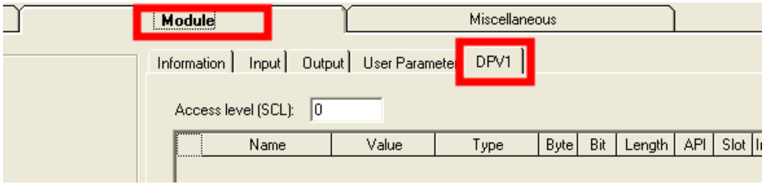
1. Start the FDT frame tool with the prepared topology tree containing the device-specific Basic PROFIBUS DTM in offline mode.
2. Open the Configuration application from the DTM menu
3. Select the **Miscellaneous** tab
4. Enable through *Enable at Device Coupler* check box the DPV1 support in the DPV1 Parameter Area. DPV1 parameter can be defined for the complete device (enable at device coupler) or for a particular module (enable at device modules).
   – If only the check box *Enable at Device Coupler* is selected, a new DPV1 tab is displayed next to the **User Parameter** tab. All included DPV1 parameter will be valid for the device type, including all modules.



Access level (SCL): 0

Figure 2. DPV1 Support at Device Coupler

- If only the check box *Enable at Device Modules* is selected, a new DPV1 tab is displayed under the **Module** tab.



Access level (SCL): 0

Figure 3. DPV1 Support at Device Modules

- If both the check boxes are selected, then the, DPV1 Parameter can be defined for the complete device and the specific module type.

5. Right-click on the application.
6. Select *Create Component from* the displayed menu.

It is possible to load or import an existing DPV1 Parameter, if required. Click on the **[Import]** button to start the import.

7. Include in the grid line the required DPV1 parameter.
8. Repeat step 5 to step 6 till all parameters are included.
9. Click on the **[Check]** button to verify, if all parameters are valid and do not have any corrupt data.
10. Click on the **[Apply]** button to confirm the parameter settings.
11. Click on the **[Export]** button to save the configuration to a file.

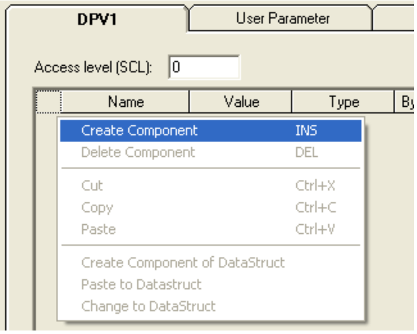
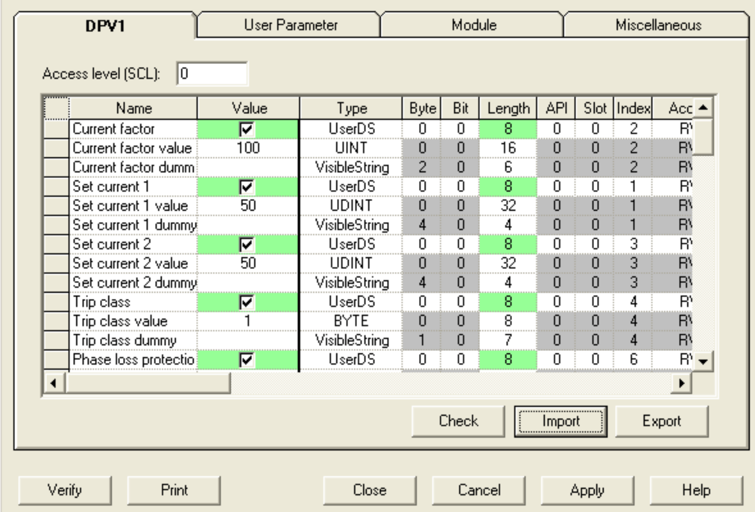


Figure 4. Create New DPV1 Parameter



Access level (SCL): 0

Name	Value	Type	Byte	Bit	Length	API	Slot	Index	Acc
Current factor	☑	UserDS	0	0	8	0	0	2	RV
Current factor value	100	UINT	0	0	16	0	0	2	RV
Current factor dumm		VisibleString	2	0	6	0	0	2	RV
Set current 1	☑	UserDS	0	0	8	0	0	1	RV
Set current 1 value	50	UDINT	0	0	32	0	0	1	RV
Set current 1 dummy		VisibleString	4	0	4	0	0	1	RV
Set current 2	☑	UserDS	0	0	8	0	0	3	RV
Set current 2 value	50	UDINT	0	0	32	0	0	3	RV
Set current 2 dummy		VisibleString	4	0	4	0	0	3	RV
Trip class	☑	UserDS	0	0	8	0	0	4	RV
Trip class value	1	BYTE	0	0	8	0	0	4	RV
Trip class dummy		VisibleString	1	0	7	0	0	4	RV
Phase loss protectio	☑	UserDS	0	0	8	0	0	6	RV

Figure 5. Included DPV1 Parameter

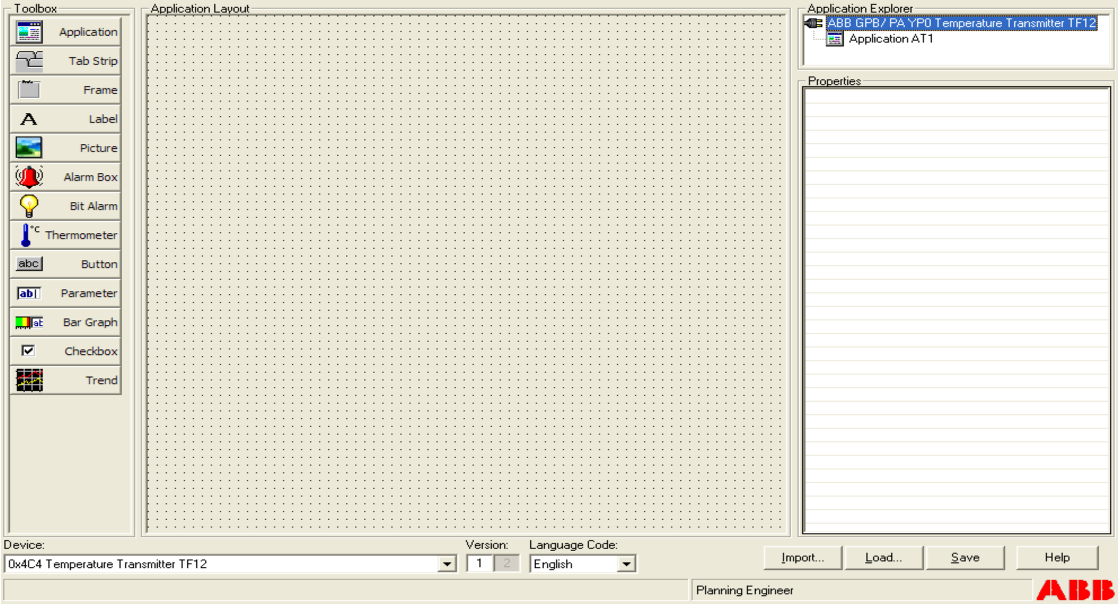
12. Close the application by clicking on the **[Close]** button.

Refer to DPV1 Configuration Application on page 61 for more details.

Open PROFIBUS Application Editor

To open the Application Editor:

1. Insert the device-specific Basic PROFIBUS DTM in the FDT/DTM project topology tree of the FDT frame tool.
2. Select the device instance in the topology tree with a right-click on the mouse button and open from the DTM menu the **Application Editor** application.



Device: 0x4C4 Temperature Transmitter TF12

Version: 1 2

Language Code: English

Figure 6. Application Editor

The Application Editor is separated into 5 areas:

- Toolbox Area.
- Application Layout Area.
- Application Explorer Area.
- Properties Area.
- Button Control Area.

The Application Layout Area is the main area to create DTM application (Graphical User Interfaces). Standard Microsoft Windows functionalities like Cut, Copy, Paste, Remove/Delete and Add/New are available for ease of work.

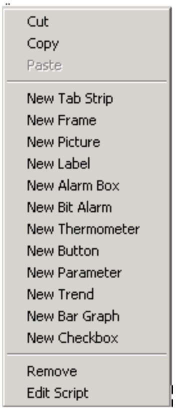
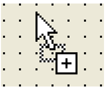


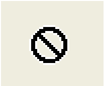
Figure 7. Application Layout Area functionalities

Controls can be inserted from the toolbox in the **Application Layout Area** or in the **Application Editor Area** through the drag'n drop functionality. It is possible to determine from the displayed mouse indicator whether a control can be dropped to a certain place or not:

Drag'n Drop allowed:



Drag'n Drop not allowed:



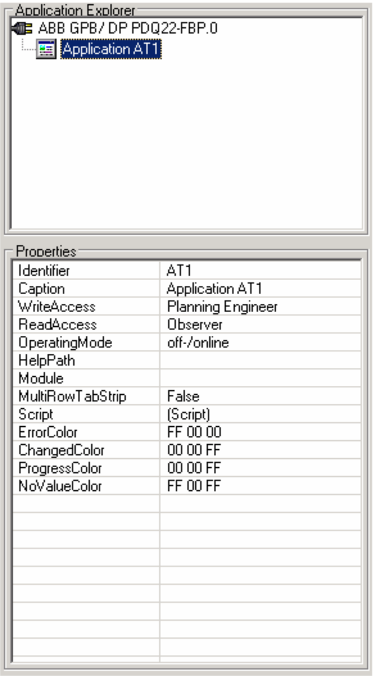
When selected, the control's position can be moved by dragging it around the layout frame or moving it to another control. Dragging of controls is represented by a frame of the control size. A control is always aligned to a grid of 8 x 8 pixels.

Create a New Application

An application is a graphical user interface provided by the DTM, which can be opened through the DTM menu by the user.

The procedure followed to create a new application is:

1. Select from the toolbox the **Application** control and move it into the **Application Explorer Area** by using the drag'n drop functionality.



Identifier	AT1
Caption	Application AT1
WriteAccess	Planning Engineer
ReadAccess	Observer
OperatingMode	off-/online
HelpPath	
Module	
MultiRowTabStrip	False
Script	(Script)
ErrorColor	FF 00 00
ChangedColor	00 00 FF
ProgressColor	00 00 FF
NoValueColor	FF 00 FF

Figure 8. New Application in the Application Explorer Area

2. Switch to the **Properties Area**. Configure and set the following parameter as required for the control:
   - Identifier
     Specify a name to identify the application.

- Caption
  Specify a name for the application, to be included in the DTM menu.
- WriteAccess
  The access rights can be configured here. All standard FDT user roles can be selected from the pull down menu. The selected user role and the user roles with higher privileges will have full control access with write permission. By default, the Planning Engineer is available. The following user roles are available according to FDT 1.2: Operator, Observer, Maintenance, Planning Engineer.
- ReadAccess
  The access rights can be configured here. All standard FDT user roles can be selected from the pull down menu. The selected user role and the user with higher privileges will have full control access with read permission. By default, the Observer is available. The following user roles are available according to FDT 1.2: Operator, Observer, Maintenance, Planning Engineer.
- Operating Mode
  Select the operating mode:
  -> Offline means, that this application can be started in offline mode only
  -> Online means, that this application can be started in online mode only.
  -> Off-/Online means to open the application in both modes.
- HelpPath
  If a specific help file (*.chm) is available, the path to a help file can be included in HelpPath.
- Module
  The Module entry allows to configure a specific module type (read from the GSD file). This DTM application will be available only if the specific module is chosen here.
- MultiRow Tabstrip
  The MultiRow Tabstrip is used for setting the area of the control to be narrower than the tabstrip. If the MultiRow Tabstrip is *true*, then the tabs will be automatically distributed along as many rows as required. By default, it is *false* and the tabstrips are arranged horizontally.

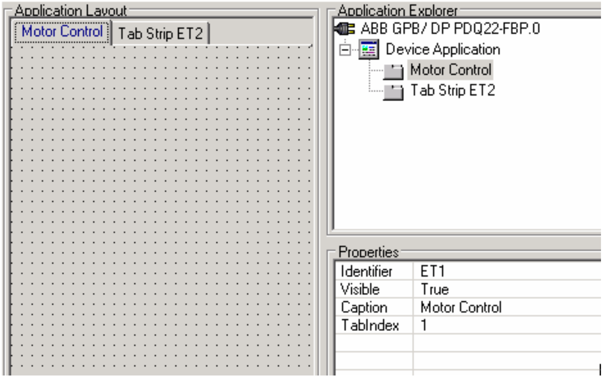
- Script
  The logical instructions applied on controls to achieve the sequence of expected behaviors. For more details, refer to Application Script Editor Area.
- ErrorColor
  Controls from the tool box and the controls used in this application will be changed to the configured color, in case of any problems. This could happen for communication errors as well as invalid configuration parameters entered by the user.
- ChangedColor
  Values or parameters, which are changed in this application by the user will be displayed in the configured color.
- ProgressColor
  Values that are currently read from or written to the device appear in this color.
- NoValueColor
  Controls from the tool box and the controls used in this application will be changed to the configured color, in case the that parameter do not have values or if any general errors occur. Additionally, this color is used as alternative error color to separate it from color used in other controls for alarm handling (e.g. BitAlarm -> true value color).

3. Repeat the steps for additional applications, if required.
4. Click on the **[Save]** button to save the application to a file, if required.

Insert new Tab Strip Control

Tab strips allow the user to create sub-applications in parallel on the same DTM master-application. A newly created sub-application can be opened by selecting its tab.

1. Select from the toolbox the **Tab Strip** control and move it into the **Application Layout Area** by using the drag'n drop functionality.



Identifier	ET1
Visible	True
Caption	Motor Control
TabIndex	1

Figure 9. Tab Strips in the Application Editor

2. Switch to the Properties Area. Set the following parameters as required for the control:

   – Identifier
   Specify a name to identify the control.

   – Visible
   Select *True* to display the control or *False* to keep it hidden.

   – Caption
   Specify the caption of the tab in this application.

   – TabIndex
   Specify the order of appearance of this tab in the Tabstrip.

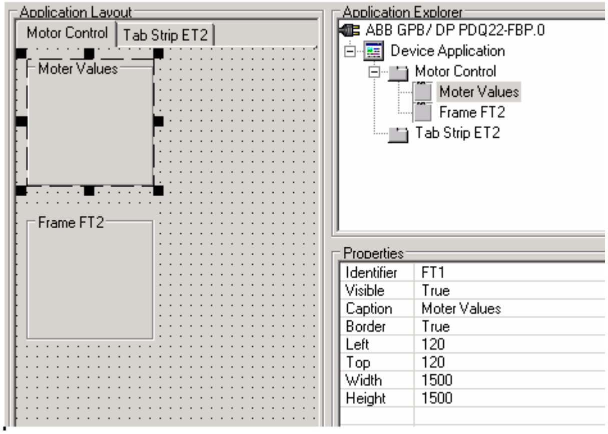
   Repeat the steps for additional tabs, if required.

3. Click on the **[Save]** button to save the application to a file.

Insert new Frame Control

Functions can be grouped together within a frame, thereby separating them from other functions. Different frames can be placed on the same Tab or Application.

1. Select from the toolbox the **Frame** control and move it into the **Application Layout Area** by using the drag'n drop functionality.



Identifier	FT1
Visible	True
Caption	Moter Values
Border	True
Left	120
Top	120
Width	1500
Height	1500

Figure 10. Frames in the Application Editor

2. Switch to the Properties Area. Configure and set the following parameter as required for the control:

   – Identifier
   Specify a name to identify the control.

   – Visible
   Select *True* to display the control or *False* to keep it hidden.

   – Caption
   Specify a caption for the frame in this application.

   – Border
   Specify as *True* to give the frame a border line. Otherwise, specify as *False*.

- Dimension
  Left and Top - Specify the position of the frame using these parameters.

  Width and Height - Specify the dimensions of the frame using these.

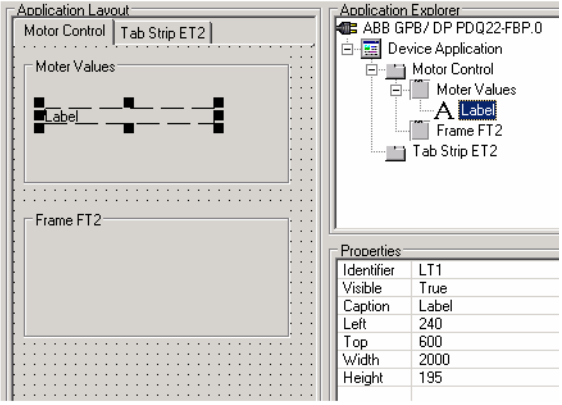
Repeat the steps for additional frames, if required.

3. Click on the **[Save]** button to save the application to a file.

Insert new Label Control

Labels allow to include an additional description in the DTM application.

1. Select from the toolbox the **Label** control and move it into the **Application Layout Area** by using the drag'n drop functionality.



Identifier	LT1
Visible	True
Caption	Label
Left	240
Top	600
Width	2000
Height	195

Figure 11. Label in the Application Editor

2. Switch to the Properties Area. Configure and set the following parameter as required for the control:

   – Identifier
   Specify a name to identify the control.

   – Visible
   Select True to display the control or False to keep it hidden.

   – Caption
   Specify the name of the label in this application.

   – "Dimension"
   Left and Top - Specify the position of the frame using these parameters

   Width and Height - Specify the dimensions of the frame using these.

   Repeat the steps for additional labels, if required.

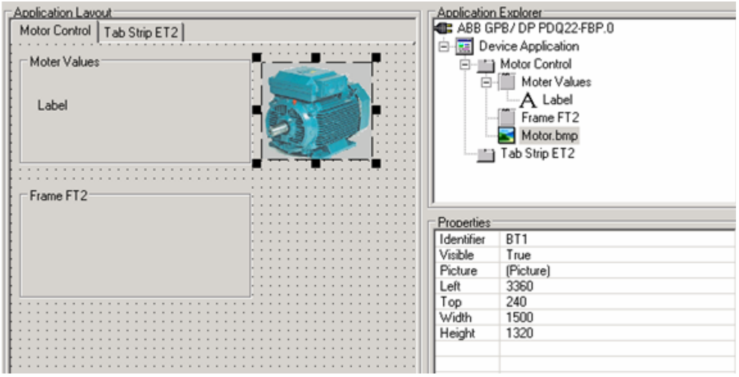
3. Click on the **[Save]** button to save the application to a file.

Insert new Picture Control

Pictures can be inserted to present more information in a graphical way to the user. Picture controls can be placed in the following tool box controls:

- Tab Strip.
- Frame.

1. Select from the toolbox the **Parameter** control and move it into the **Application Layout Area** by using the drag and drop functionality.



Identifier	BT1
Visible	True
Picture	(Picture)
Left	3360
Top	240
Width	1500
Height	1320

Figure 12. Picture Control in the Application Editor

2. Switch to the Properties Area. Configure and set the following parameters as required for the control:

   – Identifier
   Specify a name to identify the control.

   – Visible
   Select *True* to display the control or *False* to keep it hidden.

   – Picture
   Click on this property field to browse for the required picture. Select the picture file and click on the **[Open]** button. The selected picture is inserted in the **Application Layout Area**.

- “Dimension”
  Left and Top - Specify the position of the picture frame using these parameters

  Width and Height - Specify the dimensions of the picture frame using these.

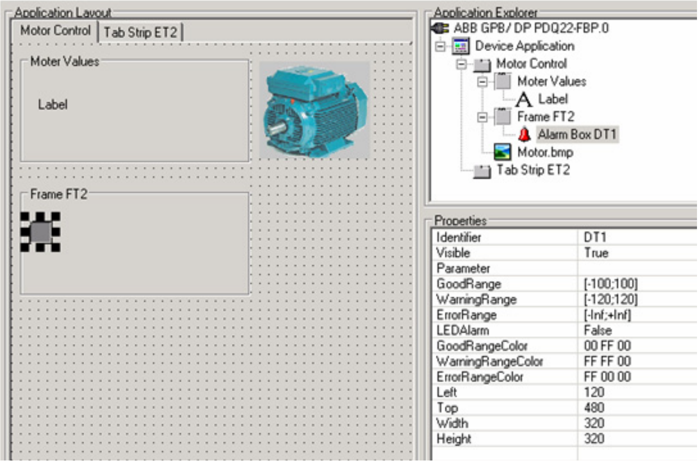
Resize the picture frame, if required.

3. Click on the **[Save]** button to save the application to a file.

Insert new Alarm Box Control

The Alarm Box control allows to include alarm notification based on a specific DPV1 parameter. If it is required that the alarm/event appear on a single bit definition only, then use the Bit Alarm instead.

1. Select from the toolbox the **Alarm Box** control and move it into the **Application Layout Area** by using the drag’n drop functionality.



Identifier	DT1
Visible	True
Parameter	
GoodRange	[-100;100]
WarningRange	[-120;120]
ErrorRange	[-Inf;+Inf]
LEDAlarm	False
GoodRangeColor	00 FF 00
WarningRangeColor	FF FF 00
ErrorRangeColor	FF 00 00
Left	120
Top	480
Width	320
Height	320

Figure 13. Alarm Box in the Application Editor

2. Switch to the **Properties Area**. Configure and set the parameter as required for the control:

   – Identifier
   Specify a name to identify the control.

   – Visible
   Select *True* to display the control or *False* to keep it hidden.

   – Parameter
   Select the DPV1 parameter, which shall be polled for an alarm/event. A pull down menu shows all configured DPV1 parameters defined in the DPV1 tab of the Configuration application.

- GoodRange
  If the DPV1 falls in the range specified here, then the alarm box shows the color defined in the property *GoodRangeColor*
- WarningRange
  If the DPV1 parameter falls in the range specified here, then the alarm box shows the color defined in the property *WarningRangeColor*
- ErrorRange
  If the DPV1 parameter falls in the range specified here, then the alarm box shows the color defined in the property *ErrorRangeColor*
- LEDAlarm
  Select *True* to have an LED style alarm notification layout, or *False* to have a box style layout.
- GoodRangeColor
  Defines the color of the alarm control between the lower warning color and the upper warning color.
- WarningRangeColor
  Defines the color of the alarm control for the value configured in *WarningRange*.
- ErrorRangeColor
  Defines the color of the alarm control for the value configured in *ErrorRange*.
- Dimension
  Left and Top - Specify the position of the frame using these parameters

  Width and Height - Specify the dimensions of the frame using these.

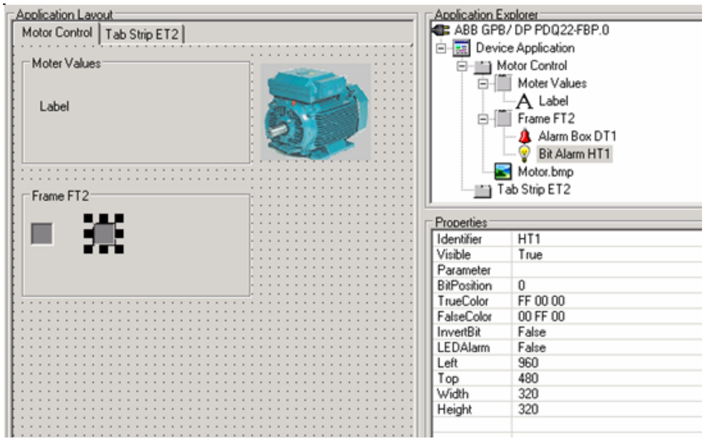
Repeat the steps for additional alarm boxes, if required.

3. Click on the **[Save]** button to save the application to a file.

Insert new Bit Alarm Control

The Bit Alarm control allows to include alarm notification based on a specific DPV1 parameter value and certain bit positions.

1. Select from the toolbox the **Bit Alarm** control and move it into the **Application Layout Area** by using the drag'n drop functionality.



Identifier	HT1
Visible	True
Parameter	
BitPosition	0
TrueColor	FF 00 00
FalseColor	00 FF 00
InvertBit	False
LEDAlarm	False
Left	960
Top	480
Width	320
Height	320

Figure 14. Bit Alarm in the Application Editor

2. Switch to the **Properties Area**. Configure and set the following parameters as required for the control:

   – Identifier
   Specify a name to identify the control.

   – Visible
   Select True to display the control or False to keep it hidden.

   – Parameter
   Select the DPV1 parameter, which shall be polled for an alarm/event. A pull down menu shows all configured DPV1 parameters defined in the DPV1 Tab of the Configuration application.

   – Bit Position
   Defines the bit position of the DPV1 parameter, which shall be monitored.

- True Color/False Color
  Depending in the bit value (1 or 0) the defined color will be displayed.
- InvertBit
  False entry handles the value *1* as checked.
  True entry handles the value *1* as unchecked.
- LEDAlarm
  Select *True* to have an LED style alarm notification layout, or *False* to have a box style layout.
- Dimension
  Left and Top - Specify the position of the frame using these parameters.

  Width and Height - Specify the dimensions of the frame using these parameters.

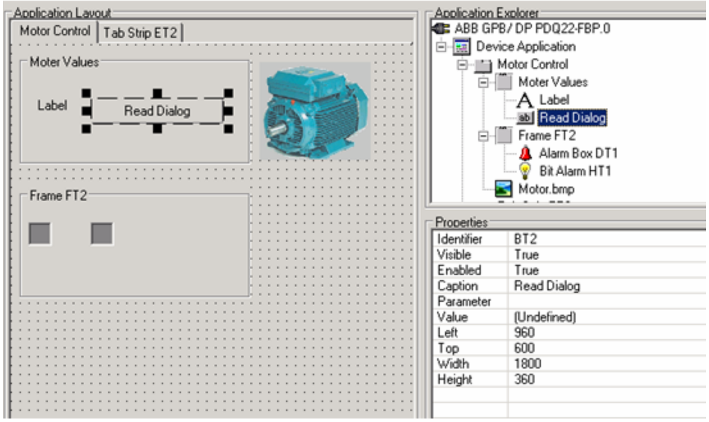
Repeat the steps for additional bit alarms, if required.

3. Click on the **[Save]** button to save the application to a file.

Insert new Button Control

A Button allows to execute specific commands defined through the DPV1 parameter manually by the user.

1. Select from the toolbox the **Button** control and move it into the **Application Layout Area** by using the drag'n drop functionality.



Application Layout

Motor Control | Tab Strip ET2

Moter Values

Label | Read Dialog

Frame FT2

Application Explorer

- ABB GPB/ DP PDQ22-FBP.0
  - Device Application
    - Motor Control
      - Moter Values
        - Label
        - Read Dialog
      - Frame FT2
        - Alarm Box DT1
        - Bit Alarm HT1
      - Motor.bmp

Properties

Identifier	BT2
Visible	True
Enabled	True
Caption	Read Dialog
Parameter	
Value	(Undefined)
Left	960
Top	600
Width	1800
Height	360

Figure 15. Button Control in the Application Editor

2. Switch to the **Properties Area**. Configure and set the following parameter as required for the control:

   – Identifier
   Specify a name to identify the control.

   – Visible
   Select True to display the control or False to keep it hidden.

   – Enabled
   The control activates when the value is *true* and disabled when the value is *false*. By default, the value is *true*.

   – Caption
   Specify the label of the button to be displayed in the application.

- Parameter
  Select the DPV1 parameter, which shall be executed by clicking the button. A pull down menu shows all configured DPV1 parameters defined in the DPV1 tab of the Configuration application.
- Value
  The value entered here will be sent to the device (via DPV1 Parameter). The value must have the correct syntax of the required data type.
- Dimension
  Left and Top - Specify the position of the button using these parameters.

  Width and Height - Specify the dimensions of the button using these.

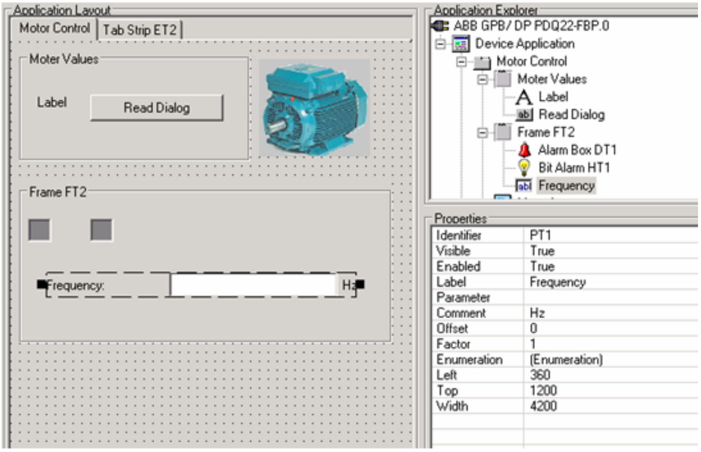
Repeat the steps for additional buttons, if required.

3. Click on the **[Save]** button to save the application to a file.

Insert new Parameter Control

The Parameter control can be used as combo box or as list box.

1. Select from the toolbox the **Parameter** control and move it into the **Application Layout Area** by using the drag'n drop functionality.

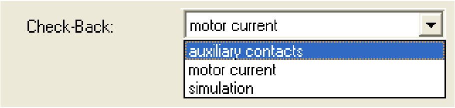


Identifier	PT1
Visible	True
Enabled	True
Label	Frequency
Parameter	
Comment	Hz
Offset	0
Factor	1
Enumeration	(Enumeration)
Left	360
Top	1200
Width	4200

Figure 16. Parameter Control in the Application Editor

2. Switch to the **Properties Area**. Configure and set the parameter as required for the control:
   (Combo Box and List Box is described separately)

Create a Combo Box

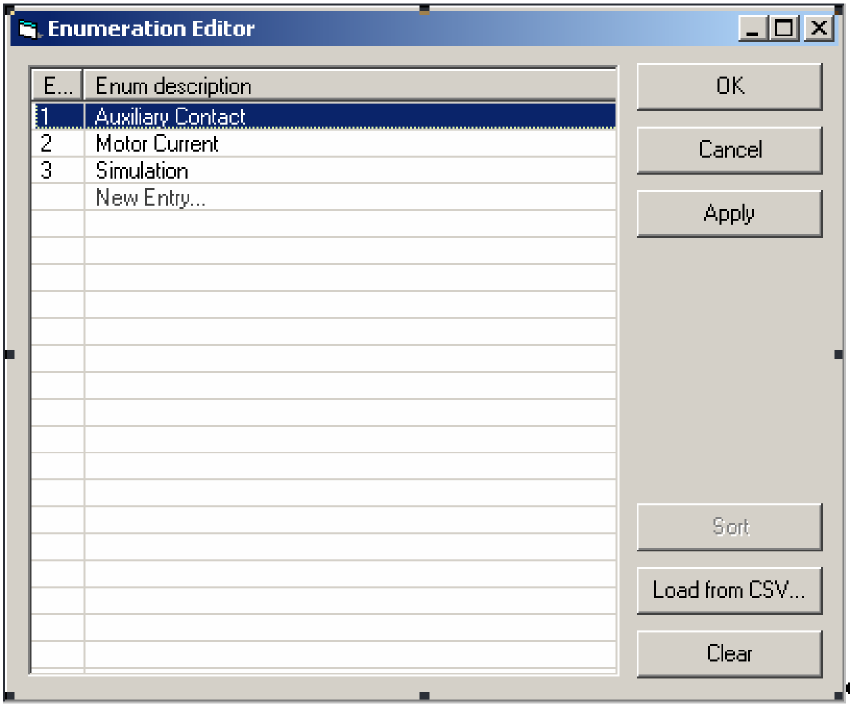


Check-Back: motor current

Figure 17. Combo box with Pull Down Menu

- Identifier
  The name of the application that is used to identify the application.
- Visible
  The Visible is used to display or hide the control.
- Enabled
  The control activates when the value is *true* and disabled when the value is *false*. By default, the value is *true*.
- Label
  Insert here the name for the Parameter control, which is visible in the application.
- Parameter
  Select the DPV1 parameter, which shall be displayed or edited. A pull down menu shows all configured DPV1 parameter defined in the DPV1 tab of the Configuration application.
- Comment
  Include here a comment, which is visible next to the control, for example, the unit, if applicable.
- Offset
  The Offset value is added to the raw value defined in the parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.
- Factor
  The value included in the factor property is multiplied with the raw value defined in the parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.

– Enumeration
Click on this property field and on the **[?]** button to open an editor to define a value set, which will be displayed and selectable in the DTM application as pull down menu. Only the pre-configured values can be selected by the user. The parameter must be defined as (unsigned) integer data type.
The opened editor allows to reuse pre-defined values by clicking on the [**Load from CSV]** button, if applicable. If a CSV file is not available, the enumerations have to be included in the grid manually.



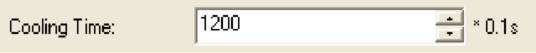
E...	Enum description
1	Auxiliary Contact
2	Motor Current
3	Simulation
	New Entry...

Figure 18. Enumeration Editor window

Confirm the enumeration configuration with the [**OK]** button or [**Apply]** button. The [**OK**] button closes the editor and opens the **Application Editor**.

- “Dimension”
  The property entries’ Left, Top, Width, Height’ can be used to adjust the button frame to a certain size.

Create a List Box



Cooling Time: 1200 * 0.1s

Figure 19. List Box

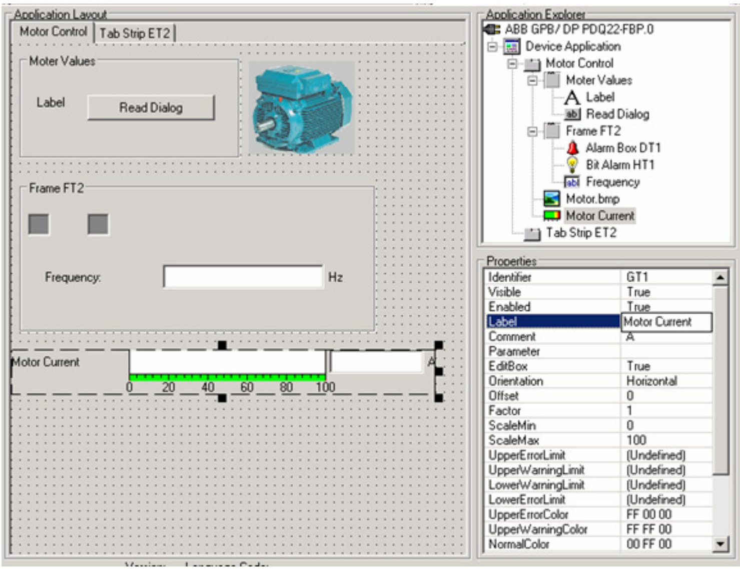
- Label
  Insert here the name for the Parameter control, which is visible in the application.
- Parameter
  Select the DPV1 parameter, which shall be sent to the device. A pull down menu shows all configured DPV1 parameter defined in the DPV1 tab of the Configuration application. The DPV1 parameter in the DPV1 parameter list must have *Min* and *Max* values configured. Parameter type string is not permitted.
- Comment
  Include here a comment, which is visible next to the control, for example, the unit, if applicable.
- Offset
  The Offset value is added to the raw value defined in the parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.
- Factor
  The value included in the factor property is multiplied with the raw value defined in the parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.
- Enumeration
  Configuration of enumeration is not required.
- “Dimension”
  The property entries’ Left, Top, Width, Height’ can be used to adjust the button frame to a certain size.

3. Repeat the steps for additional parameter controls, if required.
4. Click on the **[Save]** button to save the application to a file, if required.

Insert new Bar Graph Control

Bar Graph control allows to visualize measured value in a graphical way. It is for numeric data types only. Additionally, to the graphical display the measured value is shown as real value next to the bar graph.

1. Select from the toolbox the **Bar Graph** control and move it into the **Application Layout Area** by using drag’n drop functionality.



Identifier	GT1
Visible	True
Enabled	True
Label	Motor Current
Comment	A
Parameter	
EditBox	True
Orientation	Horizontal
Offset	0
Factor	1
ScaleMin	0
ScaleMax	100
UpperErrorLimit	(Undefined)
UpperWarningLimit	(Undefined)
LowerWarningLimit	(Undefined)
LowerErrorLimit	(Undefined)
UpperErrorColor	FF 00 00
UpperWarningColor	FF FF 00
NormalColor	00 FF 00

Figure 20. Bar Graph in the Application Editor

2. Switch to the Properties Area. Configure and set the parameter as required for the control:

   – Identifier
   The name of the application that is used to identify the application.

   – Visible
   The Visible is used to display or hide the control.

- Enabled
  The control activates when the value is *true* and disabled when the value is *false*. By default, the value is *true*.
- Label
  Insert here the name of the measured value, which is visible next to the bar graph.
- Comment
  Include here e.g. the unit, if applicable.
- Parameter
  Select the DPV1 parameter, for which the value shall be displayed. A pull down menu shows all configured DPV1 parameter defined in the DPV1 tab of the Configuration application.
- Edit Box
  The Edit Box (Value) field will be visible when the value is *true* and invisible when the value is *false*. By default, the value is *true*.
- Orientation
  Orientation will display the control in Horizontal or Vertical.
- Offset
  The Offset value is added to the raw value defined in the parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.
- Factor
  The value included in the factor property is multiplied with the raw value defined in the parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.
- ScaleMin
  Insert here the lowest value on the meter bar ruler.
- ScaleMax
  Insert here the highest value on the meter bar ruler.
- UpperErrorLimit
  Define, where the upper error limit starts for the DPV1 parameter. The color for this range is defined in the property UpperErrorColor.

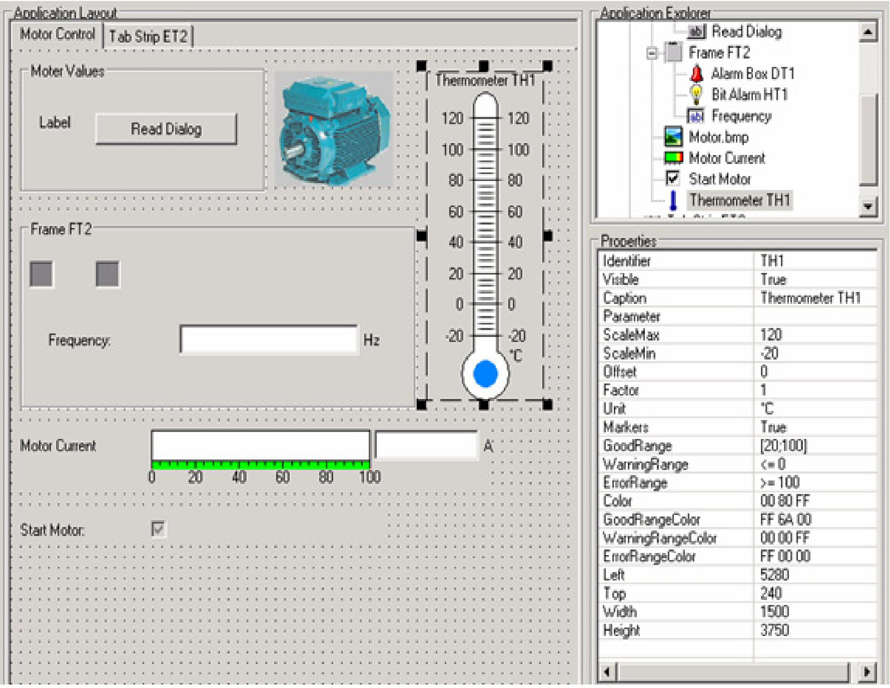
- UpperWarningLimit
  Define, where the upper warning limit starts for the DPV1 parameter. The color for this range is defined in the property UpperWarningColor
- LowerWarningLimit
  Define, where the lower warning limit starts for the DPV1 parameter. The color for this range is defined in the property LowerWarningColor.
- LowerErrorLimit
  Define, where the lower error limit starts for the DPV1 parameter. The color for this range is defined in the property LowerErrorColor.
- NormalColor
  Defines the color of the bar graph between the lower warning color and the upper warning color.
- UpperErrorColor
  Value above UpperErrorLimit appear in the color of UpperErrorColor.
- UpperWarningColor
  Value above UpperWarningLimit and below UpperErrorLimit appear in the color of UpperWarningColor.
- “Dimension”
  The property entries’ Left, Top, Width’ can be used to adjust the bar graph frame to a certain size.

3. Repeat the steps for additional bar graphs, if required.
4. Click on the **[Save]** button to save the application to a file, if required.

Insert new Thermometer Control

The Thermometer control allows to visualize measured value in an graphical way. It is for numeric data types only.

1. Select from the toolbox the **Thermometer** control and move it into the **Application Layout Area** by using the drag'n drop functionality.



Identifier	TH1
Visible	True
Caption	Thermometer TH1
Parameter	
ScaleMax	120
ScaleMin	-20
Offset	0
Factor	1
Unit	°C
Markers	True
GoodRange	[20;100]
WarningRange	<= 0
ErrorRange	>= 100
Color	00 80 FF
GoodRangeColor	FF 6A 00
WarningRangeColor	00 00 FF
ErrorRangeColor	FF 00 00
Left	5280
Top	240
Width	1500
Height	3750

Figure 21. Thermometer Control in the Application Editor

2. Switch to the **Properties Area**. Configure and set the parameter as required for the control:

   – Identifier
   The name of the application that is used to identify the application.

- Visible
  The Visible is used to display or hide the control.
- Param<number>_Parameter
  Select the DPV1 parameter, for which the value shall be displayed. A pull down menu shows all configured DPV1 parameter defined in the DPV1 tab of the Configuration application.
- Param<number>_Factor
  The value included in the factor property is multiplied with the raw value defined in the parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.
- Param<number>_Offset
  The Offset value is added to the raw value defined in the parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.
- Param<number>_Scale Min
  Insert here the lowest value on the thermometer bar ruler.
- Param1_Scale Max
  Insert here the highest value on the thermometer bar ruler.
- Param<number>_Interpolate
  This property allows to interpolate trend values between two measured values.
  -> True value enables interpolation
  -> False value disables interpolation (step curves)
- Param<number>_Unit
  Insert here the unit description, visible below the thermometer.
- Param<number>_Color
  Defines the color of the thermometer fluid, if whether *GoodRange* or *WarningRange* or *ErrorRange* is read from the process.
- Background
  Defines the background color of the trend
- LineColor
  Defines the vertical and horizontal line colors of the trend grid

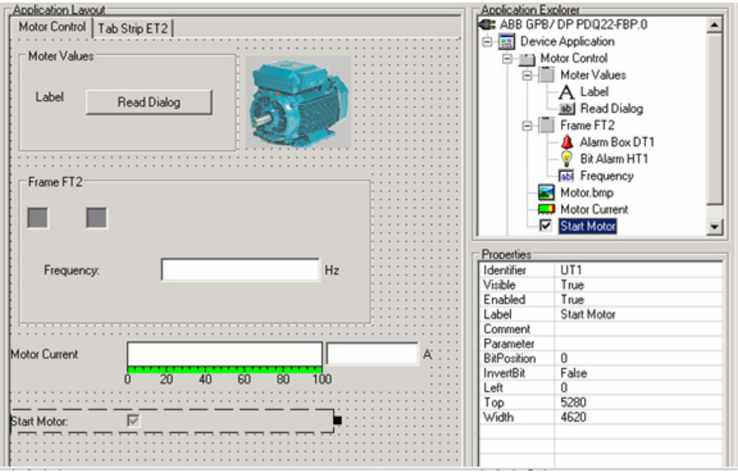
- Caption
  Insert here the name of the measured value, which is visible above the thermometer.
- "Dimension"
  The property entries' Left, Top, Width' can be used to adjust the bar graph frame to a certain size.

3. Repeat the steps for additional bar graphs, if required.
4. Click **[Save]** button to save the application to a file, if required.

Insert new Check Box Control

The Check Box control allows to sent to or read from the device a boolean value.

1. Select from the toolbox the **Bit Alarm** control and move it into the Application Layout Area by using the drag'n drop functionality.



Identifier	UT1
Visible	True
Enabled	True
Label	Start Motor
Comment	
Parameter	
BitPosition	0
InvertBit	False
Left	0
Top	5280
Width	4620

Figure 22. Check Box Control in the Application Editor

2. Switch to the **Properties Area**. Configure and set the parameter as required for the control:

   – Label
   Insert here the name for the Parameter control, which is visible in the application.

   – Comment
   Include here, for example, the unit, if applicable.

   – Parameter
   Select the DPV1 parameter, which shall be used for the check box. A pull down menu shows all configured DPV1 parameter defined in the DPV1 tab of the Configuration application.

- Bit Position
  Defines, which bit position of the DPV1 parameter shall be monitored.
- InvertBit
  False entry handles the value *1* as checked.
  True entry handles the value *1* as unchecked.
- “Dimension”
  The property entries’ Left, Top, Width’ can be used to adjust the check box frame to a certain size.

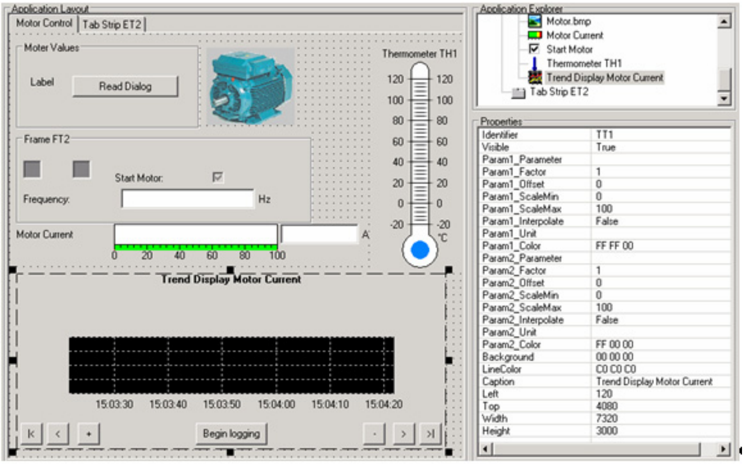
Repeat the steps for additional check boxes, if required.

3. Click on the **[Save]** button to save the application to a file.

Insert new Trend Control

The Trend control allows to visualize up to two values in parallel over a certain point of time. It is for numeric data types only. Additionally, the values can be saved in a log file.

1. Select from the toolbox the **Trend** control and move it into the **Application Layout Area** by using the drag'n drop functionality.



Identifier	TT1
Visible	True
Param1_Parameter	
Param1_Factor	1
Param1_Offset	0
Param1_ScaleMin	0
Param1_ScaleMax	100
Param1_Interpolate	False
Param1_Unit	
Param1_Color	FF FF 00
Param2_Parameter	
Param2_Factor	1
Param2_Offset	0
Param2_ScaleMin	0
Param2_ScaleMax	100
Param2_Interpolate	False
Param2_Unit	
Param2_Color	FF 00 00
Background	00 00 00
LineColor	C0 C0 C0
Caption	Trend Display Motor Current
Left	120
Top	4080
Width	7320
Height	3000

Figure 23. Trend Control in the Application Editor

2. Switch to the **Properties Area**. Configure and set the parameter as required for the control:

   – Param<number>_Parameter
   Select the DPV1 parameter, which shall be trended. A pull down menu shows all configured DPV1 parameter defined in the DPV1 tab of the Configuration application.

   – Param<number>_Offset
   The Offset value is added to the raw value defined in the parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.

- Param<number>_Factor
  The value included in the factor property is multiplied with the raw value defined in the parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.

- Param<number>_Scale Min
  Insert here the lowest value for the scale.

- Param<number>_Scale Max
  Insert here the highest value for the scale.

- Param<number>_Interpolate
  This property allows to interpolate trend values between two measured values.
  -> True value enables interpolation
  -> False value disables interpolation (step curves)

- Param<number>_Unit
  Unit of the measured value, e.g. “ml”, “kg”, “mol”

- Param<number>_Color
  This property defines the color, in which the trend line for this parameter will be visible.

- Background
  Defines the background color of the trend

- LineColor
  Defines the vertical and horizontal line colors of the trend grid

- Caption
  Include the label for the trend, displayed as header for the trend in the DTM application.

- “Dimension”
  The property entries’ Left, Top, Width’ can be used to adjust the bar graph frame to a certain size.

Repeat the steps for additional bar graphs, if required.

3. Click on the **[Save]** button to save the application to a file, if required.

Device Specific Basic PROFIBUS DTM Application

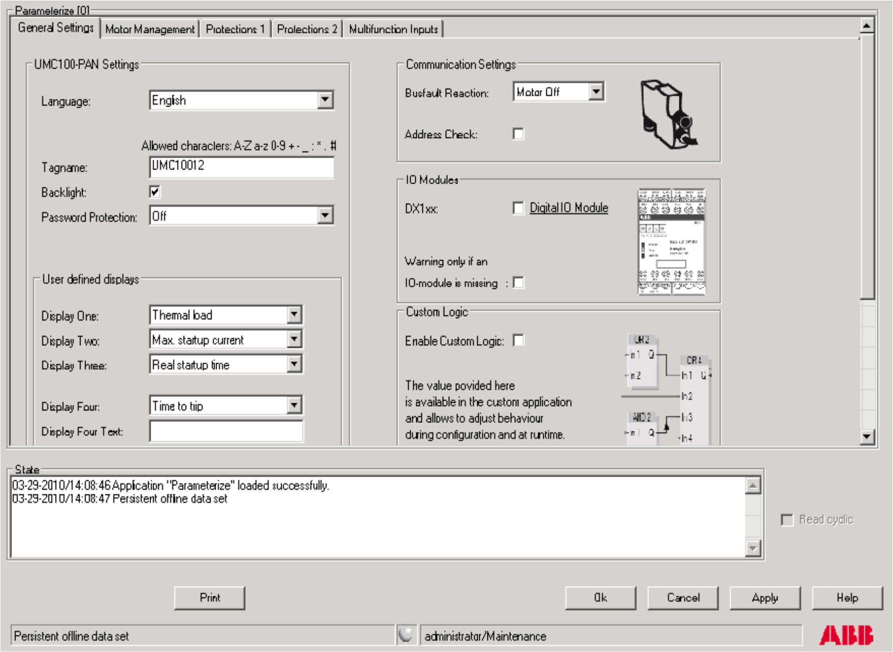
The created DTM applications can be used to parameterize and to configure connected device types, which belongs to the included GSD file. The applications are available for the user according to the user rights and the mode (online/offline) of the DTM.

The user-defined applications are stored as template files on the local disc. If the DTM shall be used on other PCs, the template files must be provided to these PCs, too. It is mandatory to export all the created files to separate xml files.

Additionally, the files can be found in the following path (default):
<Installation Path>:\Program Files\ABB Industrial IT\Engineer IT\DTM\PROFIBUS DTM Builder\bin\DTM\PROFIBUS\

During start of the DTM, for the first time, all the pre-defined templates and user applications are copied to the main path of the project (project related path) automatically.

After saving these files inside a specific device DTM these templates and user applications were made device-specific. When the DTM is called next time the DPV1 parameter and user applications are shown directly.



UMC100-PAN Settings

Language: English

Allowed characters: A-Z a-z 0-9 + - _ : * . #

Tagname: UMC10012

Backlight: ☑

Password Protection: Off

User defined displays

Display One: Thermal load

Display Two: Max. startup current

Display Three: Real startup time

Display Four: Time to trip

Display Four Text:

Communication Settings

Busfault Reaction: Motor Off

Address Check: □

IO Modules

DX1xx: □ Digital IO Module

Warning only if an IO-module is missing : □

Custom Logic

Enable Custom Logic: □

The value povided here is available in the custom application and allows to adjust behaviour during configuration and at runtime.

State

```
03-29-2010/14:08:46 Application "Parameterize" loaded successfully.
03-29-2010/14:08:47 Persistent offline data set
```

□ Read cyclic

Persistent offline data set | administrator/Maintenance

Figure 24. Example for a Created Application Accessible in Offline Mode

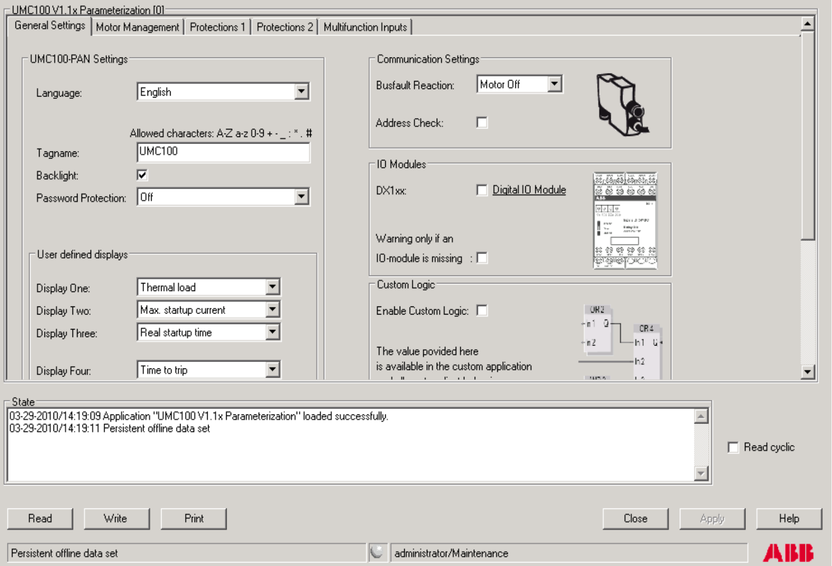
In the offline mode, the DTM application will start with the last stored values, which are stored persistent in the instance data record. The user can adapt the data to the device configuration.

The following buttons are available in offline applications:

- The **[OK]** button applies all changed values persisting in the instance data record and closes the dialog.
- The **[Cancel]** button closes the dialog without saving the data to the instance data record.
- The **[Apply]** button applies all changed values in the instance data record. The application window remains open.

- The **[Print]** button is used to print all the parameters available in the application.

To synchronize the instance data record with the device, the DTM must be changed to online mode. Writing single device data as well as a complete download can be initiated through DTM application or FDT frame tool (complete download).



Language: English

Allowed characters: A-Z a-z 0-9 + - _ : * . #

Tagname: UMC100

Backlight: ☑

Password Protection: Off

Display One: Thermal load

Display Two: Max. startup current

Display Three: Real startup time

Display Four: Time to trip

Busfault Reaction: Motor Off

Address Check: □

DX1xx: □ Digital IO Module

Warning only if an IO-module is missing : □

Enable Custom Logic: □

The value povided here is available in the custom application

State

```
03-29-2010/14:19:09 Application "UMC100 V1.1x Parameterization" loaded successfully.
03-29-2010/14:19:11 Persistent offline data set
```

□ Read cyclic

Persistent offline data set | administrator/Maintenance

Figure 25. Example for a Created Application Accessible in Online Mode

Online application allows to read data from the device or write parameter to the device by clicking the respective **[Read]** or **[Write]** button. If DPV1 parameters are read-only values, the value field is disabled.

The following buttons are available in online applications:

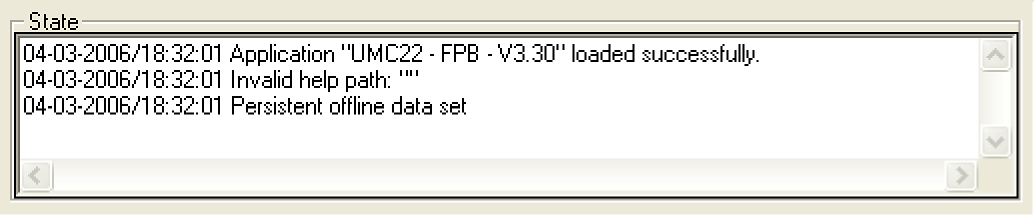
- The **[Close]** button applies all changed values persistent in the instance data record and closes the dialog. The vales are NOT written to the device.

- The **[Read]** button reads the values/parameter defined on this application from the device.

- The **[Write]** button writes the values/parameter defined on this application to the device and saves them to the instance data record.
- The **[Apply]** button applies all changed values in the instance data record. The application window remains open.
- The [**Print]** button is used to print all the parameters available in the application.

Status Message

All types of messages will be displayed in the state area. Check the messages for errors or problems.



```
04-03-2006/18:32:01 Application "UMC22 - FPB - V3.30" loaded successfully.
04-03-2006/18:32:01 Invalid help path: ""
04-03-2006/18:32:01 Persistent offline data set
```

Read Cyclic

Select this control to read the values on this application cyclically. The interval of the cyclic read is set to 5 seconds. If the update rate is not sufficient, it is recalculated and extended in each interval.

Section 3 PROFIBUS DTM Builder

General

The PROFIBUS Application Editor can be used to create new device-specific DTM application, added to the base functionality of the Basic PROFIBUS DTM. As a result, the Basic PROFIBUS DTM can be extended with DPV1 services required to read and write acyclic parameter. The GUI (Graphical User Interface) of DTM application helps the user to work easily and efficiently with the connected device.

User Roles and Access Rights

The PROFIBUS DTM Builder checks the user rights when an editor starts up. Certain users may even be barred from using the editor.

Table 1. User Roles and Access Rights

Editor	**User Roles**			
	Observer	Operator	Maintenance	Planning Engineer
Parameterization	*-/-*	*R*	*R/W*	*R/W*
Observation	*R*	*R*	*R*	*R*
Application Editor	-/-	-/-	R/W	R/W

The newly created applications cannot be included here, because access rights can be designed only through the DTM builder. Each created application can be assigned to a specific user role, which fits best to the DTM design and device access rights.

-/- = The user interface is not available for selection

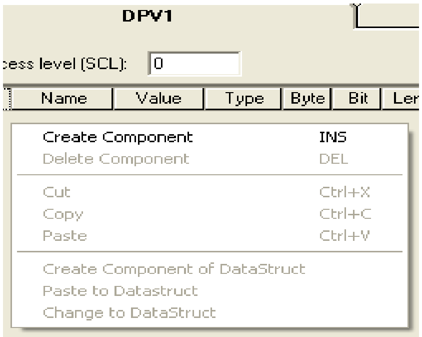
R = Data output only (read)

R/W = Data input (write) and output (read)

DPV1 Configuration Application

This application allows the user to include DPV1 parameter to be used by the DTM for acyclic communication to the connected device.

Click on this application to open a menu, where the user can select the following functions:



cess level (SCL): 0

Figure 26. Open DPV1 Menu in Configuration Application

- Create Component
  Allows to create a new line in the grid and to include specific DPV1 services.
- Delete Component
  Removes an already created DPV1 configuration (line) from the grid
- Cut, Copy, Paste
  Use this Windows functionality to multiply already created DPV1 services in the grid
- Create Component of DataStruct
  This function is usable only, if a DPV1 parameter line is set to the data type “UserDS”. Data structures for a DPV1 services can be created. Standard data structures defined in the PROFIBUS specification are already included in can be chosen from the pull down menu of data types.

- Paste DataStruct
  Data Structures can be copied by using the Copy selection and paste by using this paste function. The paste function is enabled only, if a DPV1 parameter line is created with data type “UserDS”. Alternative data structure can be copied by selecting it and by using standard copy/paste function.
- Change to DataStruct
  The existing data structure can be changed to a different data structure using this menu.

The DPV1 parameter line includes the following columns to be configured:

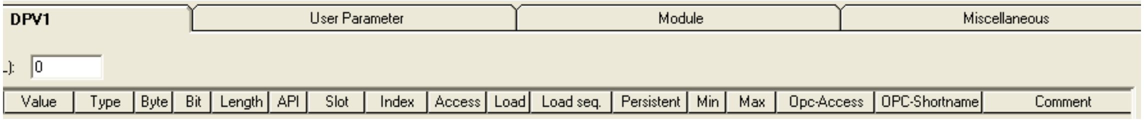


Figure 27. DPV1 Grid Columns

Table 2. DPV1 Parameter Properties

Field	Description
Name	Name of the parameter. The parameter name is visible in the Application Editor for selection. If the parameter has been defined in a module, the name of the parameter is pre-fixed with the module name when displayed in this grid. Thus, it is easy to make out the parameters belonging to a specific module. This also gives the functionality of listing the parameters module-wise when they are listed alphabetically.
Value	The value is the input/output data. This field is editable by the standard user to configure the connected device type. Default values can be set here.

Table 2. DPV1 Parameter Properties (Continued)

Field	Description
Type	This field allows the user to select the required data type for a particular DPV1 service from the pull down menu. All standard parameter and data structures are included in the list. Additional data structure can be created by using the data type “UserDS”
Byte	The Byte field defines the byte position, where the parameter value is accessible.
Bit	The Bit field defines the bit position, where the parameter value is accessible.
Length	The length field defines the length of the parameter value in the data frame, beginning from the byte/bit position.
API	This column can be used as alternative addressing to the Slot and Index addressing.
Slot	The Slot defines the position of the parameter in the device
Index	The Index is address at the slot position, where the data can be read or write.
Access	This column defines, whether the parameter can be access for read, write or read/write access.
Load	Select this field if the parameter need to be downloaded to the device by using download function [**Write**] in the DTM or through the FDT frame tool.
Load Sequence	This field allows to define a specific sequence in which the parameter shall be downloaded to the device. Enter a sequence number for a sequenced load operation.

Table 2. DPV1 Parameter Properties (Continued)

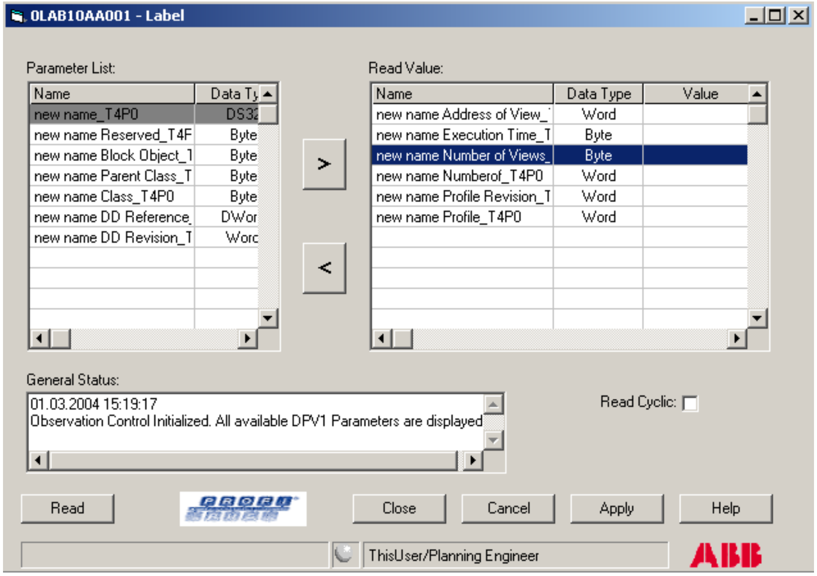
Field	Description
Persistent	Select this field if the entered value is to be persistent in the instance data record. This field should be selected, if the load column is selected, too.
Min	This field allows to enter a pre-defined minimum value. If the value is less than this included value, an error message will appear. The Min value entered here will be taken instead the invalid value.
Max	This field allows to enter a pre-defined maximum value. If the value is higher than this included value, an error message will appear. The Max value entered here will be taken instead the invalid value.
OPC Access	This option is only to be used in connection with the ABB Fieldbus OPC Server PROFIBUS/HART. If this field is selected, the parameter is available for OPC access in the OPC Server.
OPC Short Name	This field allows to define a short name for the OPC item associated to this parameter.
Comment	Comments can be included to have a more clear statement for the parameter.

Observation

Observation application allows an easy way to read the values of the DPV1 parameters from the device. The parameters that need to be read can be individually configured.

The Observation application includes parameter only, if DPV1 templates are available for the specific device type. Otherwise, this application does not provide any device access to read process values. Standard process values are available and sent through cyclic communication to the PROFIBUS master.

.



Parameter List:

Name	Data Ty
new name_T4P0	DS32
new name Reserved_T4F	Byte
new name Block Object_T	Byte
new name Parent Class_T	Byte
new name Class_T4P0	Byte
new name DD Reference_	DWor
new name DD Revision_T	Worc

Read Value:

Name	Data Type	Value
new name Address of View_	Word	
new name Execution Time_T	Byte	
new name Number of Views_	Byte	
new name Numberof_T4P0	Word	
new name Profile Revision_T	Word	
new name Profile_T4P0	Word	

General Status:

01.03.2004 15:19:17
Observation Control Initialized. All available DPV1 Parameters are displayed

Read Cyclic: □

Figure 28. Observation

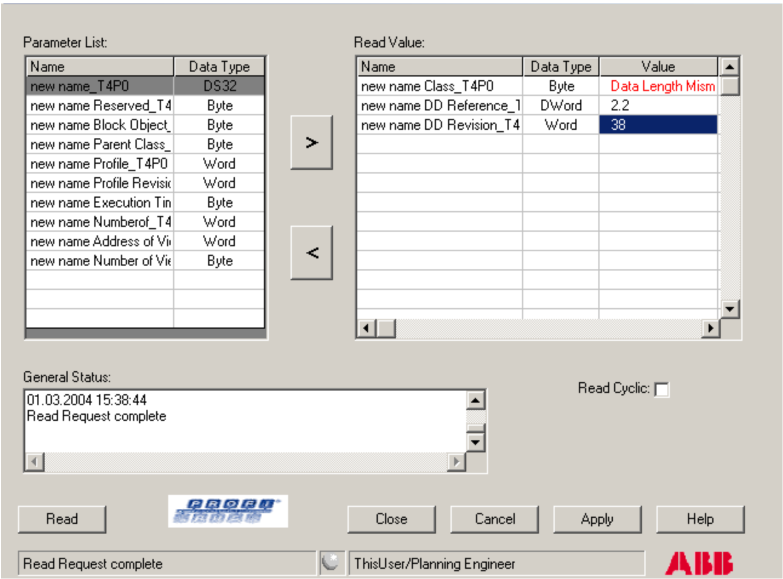
The defined DPV1 parameter list (readable parameters) is displayed on the left grid (labeled as parameter list) and can be individually moved into the right grid called *Read Valu*e using the button. No value of the data is displayed at the start up.

When “Read” or “Read Cyclically” is called, the parameters values are requested from the device through the frame application. The “General Status” box at the bottom displays the current status of the read procedure.

The box also displays any error code that is returned by the bus in case of a read error. Once the connection to the device is established, the green LED at the middle of the status bar starts flashing.

If the read cyclically check box is enabled, the selected DPV1 parameters (parameters in the right grid) for observations are read cyclically until the check box is disabled or the application is closed. The cycle for the successive reads is 5 sec. However, if the number of parameters selected is more and one read cycle is not completed in 5 seconds, the subsequent cycle is skipped without causing any errors.

For selecting multiple parameters to be shifted to the “Read Value” grid, one can press **Shift** and select multiple parameters. However, the structure headers cannot be shifted due to the fact that headers are a logical grouped and the constituent parameters are the actual parameters.



Parameter List:

Name	Data Type
new name_T4P0	DS32
new name Reserved_T4	Byte
new name Block Object_	Byte
new name Parent Class_	Byte
new name Profile_T4P0	Word
new name Profile Revisi	Word
new name Execution Tin	Byte
new name Numberof_T4	Word
new name Address of Vi	Word
new name Number of Vie	Byte

Read Value:

Name	Data Type	Value
new name Class_T4P0	Byte	Data Length Mism
new name DD Reference_1	DWord	2.2
new name DD Revision_T4	Word	38

General Status:

01.03.2004 15:38:44
Read Request complete

Read Cyclic: □

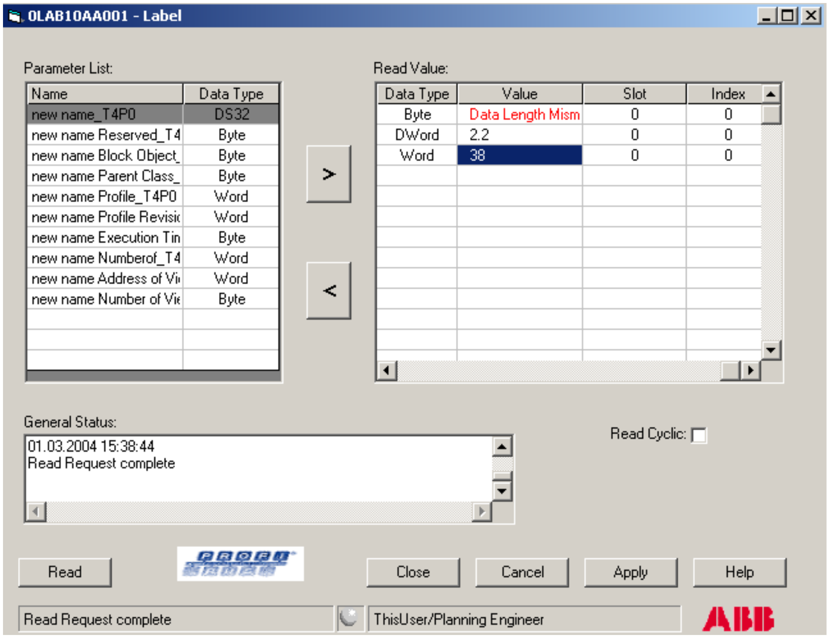
Read Request complete | ThisUser/Planning Engineer

Figure 29. Observation: Data Read

In case the configured data length does not match the length returned by the device, the grid displays the text “Data Length Mismatch” in red. The user is requested to check the configuration of the parameter. Detailed information of the specific PROFIBUS error code is displayed in the General Status box.

The Parameter List grid displays only details such as name and data type of the parameter. But, when the parameter is shifted to the Read Value grid, the other details such as slot and index values are also displayed as shown in Figure 30. Use the [<] button to shift the parameters from the Read Value grid to the Parameter List grid.

The logic takes care of arranging the parameters to their respective structure positions. The user can also save the selected list using the **[Apply]** button so that next time when the application is displayed, the parameter list remains intact. Notice that the parameter list and not the measured values are stored to the data set.



Parameter List:

Name	Data Type
new name_T4P0	DS32
new name Reserved_T4	Byte
new name Block Object_	Byte
new name Parent Class_	Byte
new name Profile_T4P0	Word
new name Profile Revisic	Word
new name Execution Tin	Byte
new name Numberof_T4	Word
new name Address of Vi	Word
new name Number of Vie	Byte

Read Value:

Data Type	Value	Slot	Index
Byte	Data Length Mism	0	0
DWord	2.2	0	0
Word	38	0	0

General Status:

01.03.2004 15:38:44
Read Request complete

Read Request complete | ThisUser/Planning Engineer

Figure 30. Observation: Parameter Attributes Displayed

The application does not allow any editing. Any changes that need to be made in the configuration have to be done in the offline phase (Configuration application).

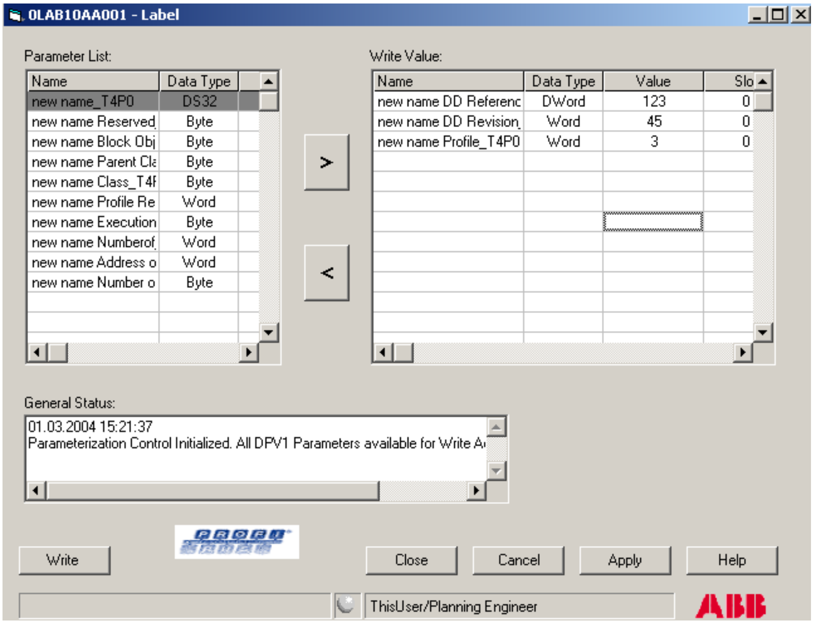
Clicking on the **[Cancel]** button will disconnect the DTM from the device and stops the cyclic read.

Parameterization

The Parametrization application includes parameter only, if DPV1 templates are available for the specific device type. Otherwise, this application does not provide any data to parametrize the connected device type.

Standard device parametrization is available and sent through cyclic communication from the PROFIBUS master to the device. If the default configuration is not sufficient for a specific device type, a DPV1 template is mandatory. Contact the device vendor for a template file or for a specific DTM.

The Parameterization application can be started in online mode only and allows the user to enter the values of the DPV1 parameters and write them to the device.



Parameter List:

Name	Data Type
new name_T4P0	DS32
new name Reserved	Byte
new name Block Obj	Byte
new name Parent Cla	Byte
new name Class_T4F	Byte
new name Profile Re	Word
new name Execution	Byte
new name Numberof	Word
new name Address o	Word
new name Number o	Byte

Write Value:

Name	Data Type	Value	Slo
new name DD Referenc	DWord	123	0
new name DD Revision	Word	45	0
new name Profile_T4P0	Word	3	0

General Status:

01.03.2004 15:21:37
Parameterization Control Initialized. All DPV1 Parameters available for Write A

Figure 31. Online Parameterization

The defined DPV1 parameter list, the writable parameters is displayed on the Parameter List grid and can be individually moved to the Write Value grid using the [**>**] button. The last value that was read from the device is displayed.

The user is allowed to edit the value. When “Write” is called, the parameter values are sent to the device through the frame application.

The General Status box at the bottom displays the current status of the write procedure. The box also displays any error code that is returned by the bus in case of a write error. Once the connection to the device is established the green LED at the middle of the status bar starts flashing.

Press **Shift** in the Parameter List grid to select multiple parameters that are to be shifted to the Write Value grid. However, the structure headers cannot be shifted due to the fact that headers are a logically grouped and the constituent parameters are the actual parameters.

When a value is to be written to the device using the write command, the value is checked for the validity, which includes checking datatype and range. The color of the text remains blue. If the write command is successful the color changes to black else it changes to red. Detailed information of the specific PROFIBUS error code is displayed in the General Status box.

The Parameter List grid displays only details such as name and data type of the parameter. But, when the parameter is shifted to the Write Value grid, the other details such as slot and index values are also displayed. Use the [**<**] button to shift the parameters from the Write Value grid to the Parameter List grid.

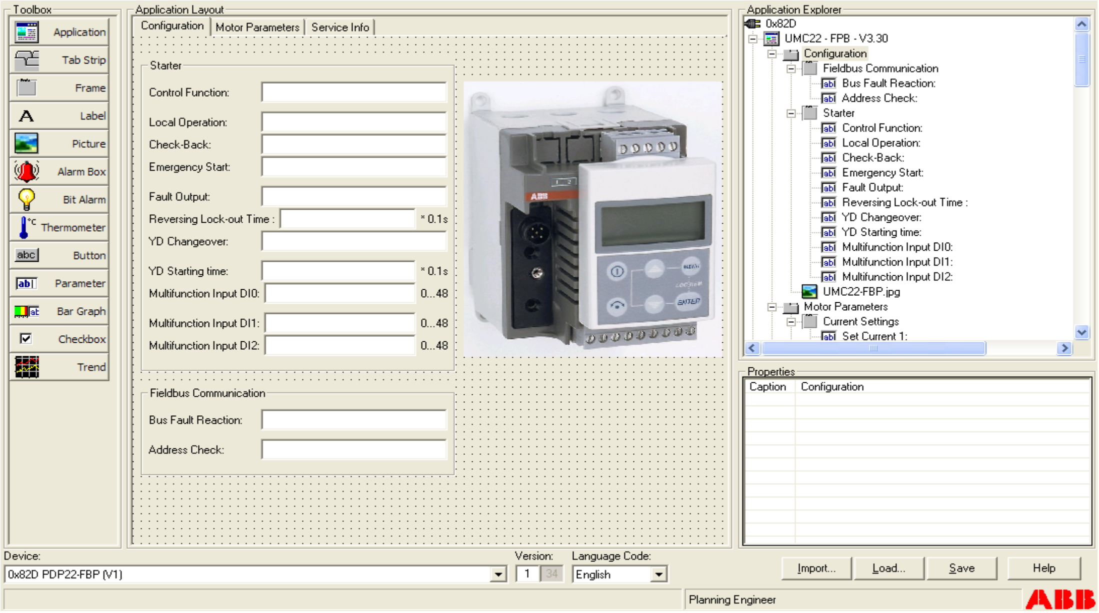
The logic takes care of arranging the parameters to their respective structure positions. The user can also save the selected list via the **[Apply]** button so that next time when the application is displayed the parameter list remains intact. Notice that the parameter list and not the values are stored to the data set. To save the device values to the instance data base and to the connected device click on the **[Write]** button.

When a write only parameter is selected for writing, all the parameters belonging to the same slot and index are selected (shifted to the right grid). Additionally, the value that needs to be written should be decimal. If the value to be written is hex it has to be prefixed with “0x” or “H”.

The application does not allow any editing. Any changes that need to be made in the configuration have to be done in offline mode via DTMs Configuration application.

PROFIBUS Application Editor

The Application Editor allows customizing the Basic PROFIBUS DTM according to the device requirements. It allows to create own Graphical User Interfaces and allocates the included DPV1 parameters to the specific DTM application. The DTM can be extended with multiple applications or with a single application that has a provision to design multiple tabs.


Device: 0x82D PDP22-FBP (V1)

Version: 1 34

Language Code: English

Caption: Configuration

Figure 32. Application Editor with Example DTM

The Application Editor window is separated into five sections:

1. The Toolbox Area on the left.
2. The Application Layout Area in the middle.
3. The Application Explorer Area at the top right.
4. The Property Editor Area at the bottom right.
5. The Button Control Area at the very bottom.

Toolbox Area

The Toolbox Area contains various controls to design an application for a specific device type. The controls are placed in the **Application Editor Area** by using the drag'n drop functionality provided by the Windows system. It is possible to use the **Application Explorer Area** as an alternative to the drag'n drop functionality of Tool Box Area. Right-click on the **Application Explorer Area** and select the required control.

The control must be moved and place manually in the Application Editor Area afterwards.

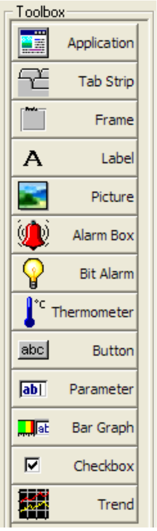
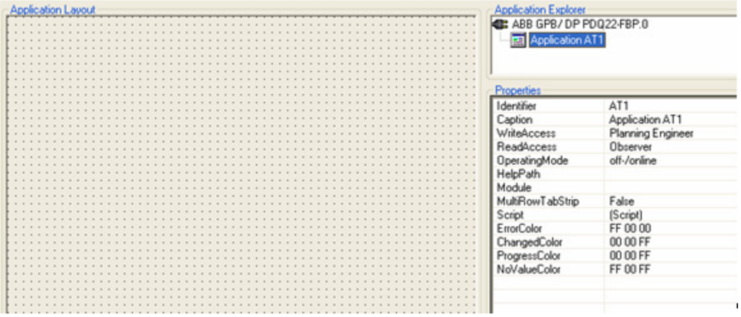


Figure 33. Toolbox Controls

Application Control

The Application control is the base of each application. It provides an empty application sheet, on which other controls can be placed. The application control cannot be placed in the Application Layout Area, but in the Application Explorer Area only.



Identifier	AT1
Caption	Application AT1
WriteAccess	Planning Engineer
ReadAccess	Observer
OperatingMode	off-/online
HelpPath	
Module	
MultiRowTabStrip	False
Script	(Script)
ErrorColor	FF 00 00
ChangedColor	00 00 FF
ProgressColor	00 00 FF
NoValueColor	FF 00 FF

Figure 34. Application Control

Properties:

The Help file has to be a “.chm” Microsoft compatible help file. The user can then use the application path or the project related path using the appl.path or project.path respectively. This way it is easy to port the application across different machines.

But, while porting the Microsoft help files (*.CHM), the files need to be transferred as well and stored in the appropriate path.

Table 3. Application Control

Field	Description
Identifier	The name of the application that is used to identify the application.
Caption	The name of the application included in the DTM menu.

Table 3. Application Control (Continued)

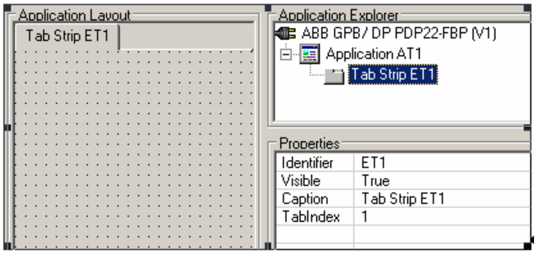
Field	Description
WriteAccess	The Access rights can be configured here. All standard FDT user roles can be selected from the pull down menu. The selected user role and higher user roles will have full control access with Write Permission. By default, the Planning Engineer is available. The following user roles are available according FDT 1.2: Operator, Observer, Maintenance, Planning Engineer.
Read Access	The Access rights can be configured here. All standard FDT user roles can be selected from the pull down menu. The selected and the higher user roles will have full control access with Read Permission. By default, the Observer is available. The following user roles are available according FDT 1.2: Operator, Observer, Maintenance, Planning Engineer.
OperatingMode	Operation mode, in which the application is shown. The following modes can be configured: online, offline or off-/online. Offline means, that the application can be started in offline mode only Online means, that the application can be started in online mode only. Off-/Online means to open the application in both modes.
HelpPath	If a specific help file (*.chm) is available the path to a help file can be included in HelpPath. By default, each application includes a Help button. If this button is clicked, the assigned help file gets opened. If there is no help file available or configured, a message in the status field of the DTM application is shown.
Module	The Module entry allows to configure a specific module type (read from the GSD file). This DTM application will be available only, if the specific module is chosen here. If no module is configured here, the DTM application is available for the device coupler (gateway).

Table 3. Application Control (Continued)

Field	Description
MultiRowTabstrip	The MultiRow tabstrip is used for setting the area of the control to be narrower than the tabstrip. If the MultiRowTabstrip is *true* then the tabs will be automatically distributed along as many rows as necessary. By default, it is *false* and the tabstrips are arranged horizontally.
Script	The logical instructions applied on controls to achieve the sequence of expected behaviors. For more details, refer to Application Script Editor Area.
ErrorColor	Controls from the tool box that are used in this application will be changed to the configured color in case of any problems. This could happen for communication errors or due to invalid configuration parameter entered by the user.
ChangedColor	Values or parameter, which are changed in this application by the user will be displayed in the configured color.
ProgressColor	Values that are currently read from or written to the device appear in this color.
NoValueColor	Controls from the tool box that are used in this application will be changed to the configured color in case the parameter do not have values or if any general errors occurs. Additionally this color is used as alternative error color to separate it from color used in other controls for alarm handling (e.g. BitAlarm -> true value color).

Tab Strip Control

Tab strips allows to create sub-application, in parallel, on the same DTM master-application. The sub-application can be opened by selecting the particular tab.



Identifier	ET1
Visible	True
Caption	Tab Strip ET1
TabIndex	1

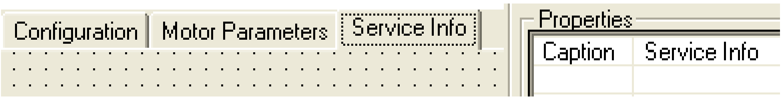
Figure 35. Tab Strip Control

Properties:

Table 4. Tab Strip Control

Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Caption	The name of the tab in this application.
Tabindex	The value of Tabndex decides the order in which the tabstrip must be displayed.

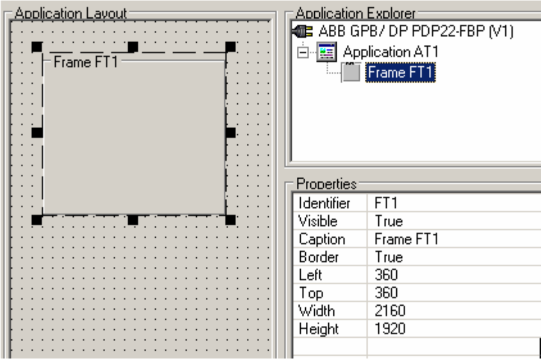
Example:



Caption	Service Info

Frame Control

The application allows to separate certain functions from other functions by grouping them into a Frame control. Different frames can be placed on the same Tab or Application.



Identifier	FT1
Visible	True
Caption	Frame FT1
Border	True
Left	360
Top	360
Width	2160
Height	1920

Figure 36. Frame Control

Properties:

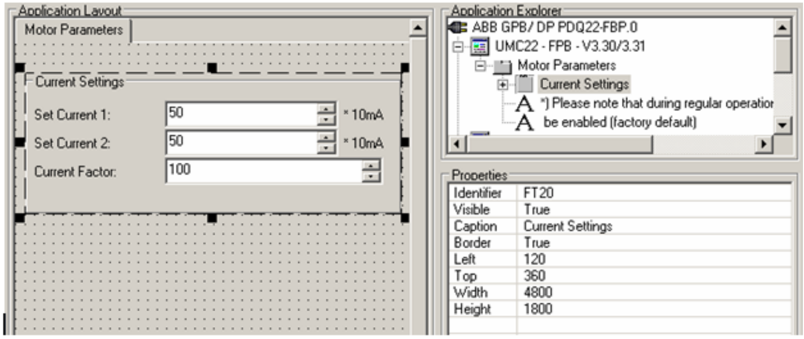
Table 5. Frame Control

Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Caption	The Caption is the name of the frame in this application.

Table 5. Frame Control (Continued)

Border	The border allows to encircle the frame with a border line. -> The border is activated with *True* value -> The border is not activated with **False** value
<u>Dimension</u> Left, Top, Width, Height	The property entries' Left, Top, Width, Height' can be used to adjust the label frame to a certain size.

Example:



Set Current 1: 50 * 10mA
Set Current 2: 50 * 10mA
Current Factor: 100

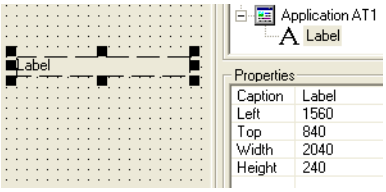
*) Please note that during regular operatio
be enabled (factory default)

Identifier	FT20
Visible	True
Caption	Current Settings
Border	True
Left	120
Top	360
Width	4800
Height	1800

Figure 37. Example for Frame Control

Label Control

Labels allows to include an additional description in the DTM application.



Properties

Caption	Label
Left	1560
Top	840
Width	2040
Height	240

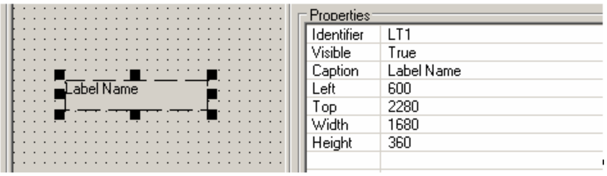
Figure 38. Label Control

Properties:

Table 6. Label Control

Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Caption	The Caption is the name of the label in this application.
<u>Dimension</u> Left, Top, Width, Height	The property entries' Left, Top, Width, Height' can be used to adjust the label frame to a certain size.

Example:

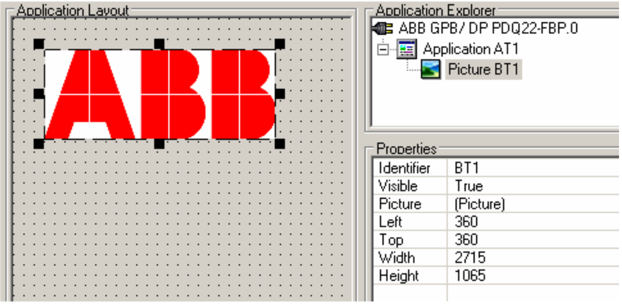


Identifier	LT1
Visible	True
Caption	Label Name
Left	600
Top	2280
Width	1680
Height	360

Figure 39. Example for Label Control

Picture Control

Pictures can be inserted to present more information in a graphical way to the user.



Identifier	BT1
Visible	True
Picture	(Picture)
Left	360
Top	360
Width	2715
Height	1065

Figure 40. Picture Control

Properties:

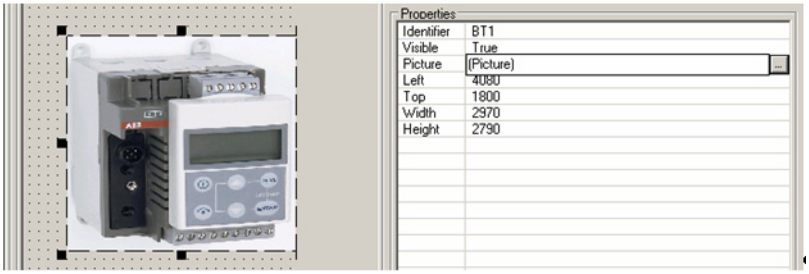
Table 7. Picture Control

Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.

Table 7. Picture Control (Continued)

Field	Description
Picture	Click, next to the Picture field, on the **[...]** button to browse for the required picture. Select the picture file and click [**Open**]. The selected picture is inserted in the Application Layout Area. Additionally the picture file is included in the application file of the DTM (<PNOID>_app.xml)
<u>Dimension</u> Left, Top, Width, Height	The property entries' Left, Top, Width, Height' can be used to adjust the label frame to a certain size.

Example:



Properties

Identifier	BT1
Visible	True
Picture	(Picture)
Left	4080
Top	1800
Width	2970
Height	2790

Figure 41. Example for Picture Control

Alarm Box Control

Alarm Box control is used it include alarm notification based on a specific DPV1 parameter.

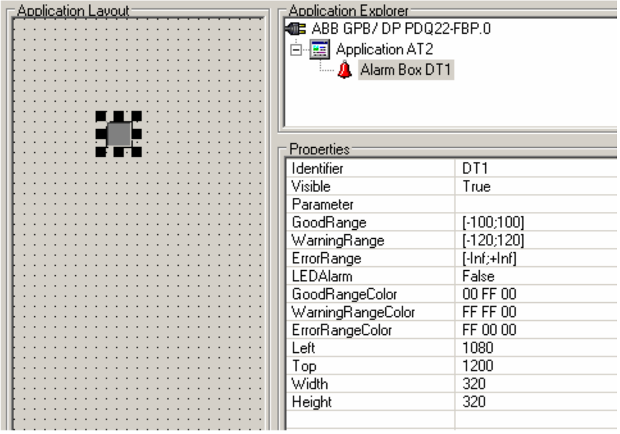


ABB GPB/ DP PDQ22-FBP.0
Application AT2
Alarm Box DT1

Identifier	DT1
Visible	True
Parameter	
GoodRange	[-100;100]
WarningRange	[-120;120]
ErrorRange	[-Inf;+Inf]
LEDAlarm	False
GoodRangeColor	00 FF 00
WarningRangeColor	FF FF 00
ErrorRangeColor	FF 00 00
Left	1080
Top	1200
Width	320
Height	320

Figure 42. Alarm Box Control

Properties:

Table 8. Alarm Box Control

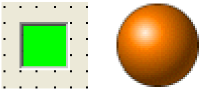
Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Parameter	The parameter used to poll the current value. A pull down menu shows all configured DPV1 parameter defined in the DPV1 Tab of the Configuration application.
GoodRange	If the value falls into this range, the box gets the color of ‘GoodRangeColor’

Table 8. Alarm Box Control (Continued)

Field	Description
WarningRange	If the value doesn’t fall into GoodRange, but into WarningRange, the box gets the color of ’WarningRangeColor’
ErrorRange	If the value is not within GoodRange or WarningRange, but within ErrorRange, the box gets the color of ’ErrorRangeColor’
LEDAlarm	The user can change the layout of the alarm notification between box style and bullet (LED) style.
GoodRangeColor	Box color for ’GoodRange’
WarningRangeColor	Box color for ’WarningRange’
ErrorRangeColor	Box color for ’ErrorRange’
Dimension Left, Top, Width, Height	The property entries’ Left, Top, Width, Height’ can be used to adjust the label frame to a certain size.

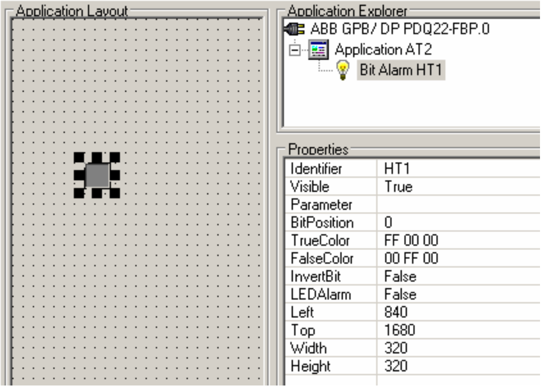
The color of the control depends on the current value of a parameter. A range is a string in the format: “(lower limit)-(upper limit)”. Ranges can overlap each other. If the value does not fall into any of the ranges, the box gets the color of the parent application’s ’NoValueColor’ property.

Example:



Bit Alarm Control

Bit Alarms allow to include an alarm notification based on a boolean value of a specific DPV1 parameter value (and certain bit position).



Identifier	HT1
Visible	True
Parameter	
BitPosition	0
TrueColor	FF 00 00
FalseColor	00 FF 00
InvertBit	False
LEDAlarm	False
Left	840
Top	1680
Width	320
Height	320

Figure 43. Bit Alarm Control

The following data types are supported:

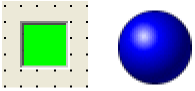
- integer.
- unsigned.
- boolean.
- OctetString.

Properties:

Table 9. Bit Alarm Control

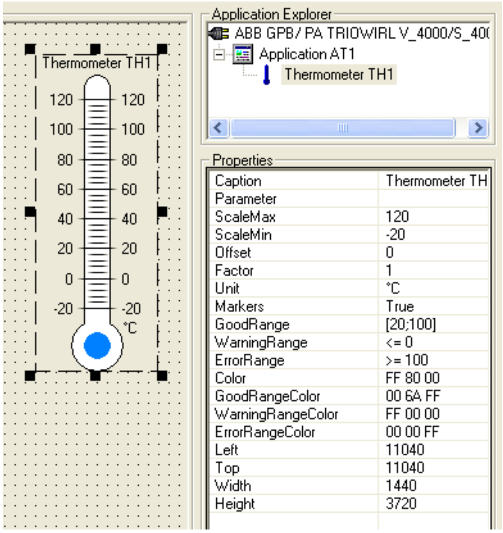
Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Parameter	The parameter used to poll the current value. A pull down menu shows all configured DPV1 parameter defined in the DPV1 Tab of the Configuration application.
BitPosition	Bit number ’0’ means least significant bit. Use ’0’ for values of a type boolean.
TrueColor	Box color if bit is ’1’
FalseColor	Box color if bit is ’0’
InvertBit	Exchanges the effect of ’0’ and ’1’ (In effect if switches TrueColor and FalseColor)
LEDAlarm	Same effect as the Alarm Box, shows a sphere instead of a box.
<u>Dimension</u> Left, Top, Width, Height	The property entries’ Left, Top, Width, Height’ can be used to adjust the label frame to a certain size.

Example:



Thermometer Control

Thermometer Control allows to visualize measured value in a graphical way. It is for numeric data types only. Additionally, for the graphical display the measured value is shown as real value next to the bar graph.



Caption	Thermometer TH
Parameter	
ScaleMax	120
ScaleMin	-20
Offset	0
Factor	1
Unit	°C
Markers	True
GoodRange	[20;100]
WarningRange	<= 0
ErrorRange	>= 100
Color	FF 80 00
GoodRangeColor	00 6A FF
WarningRangeColor	FF 00 00
ErrorRangeColor	00 00 FF
Left	11040
Top	11040
Width	1440
Height	3720

Figure 44. Thermometer Control

Properties:

Table 10. Thermometer Control

Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Caption	Name of the thermometer control.

Table 10. Thermometer Control (Continued)

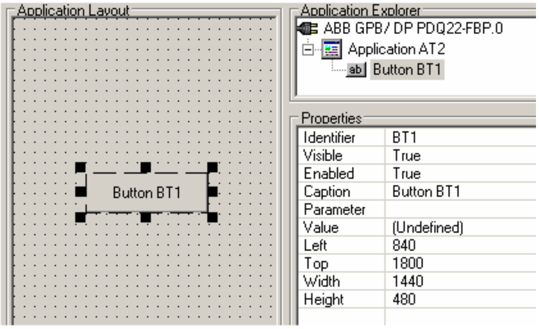
Field	Description
Parameter	Parameter value, which shall be visible. A pull down menu shows all configured DPV1 parameter defined in the DPV1 Tab of the Configuration application. Value on the thermometer bar = Offset + Factor * Parameter Value.
ScaleMin	Lowest value on the thermometer bar ruler.
ScaleMax	Highest value on the thermometer bar ruler.
Offset	The Offset value is added to the raw value defined in the parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.
Factor	The value included in the factor property is multiplied with the raw value defined in the parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.
Unit	The unit description is visible below the thermometer.
Markers	This entry allows to show or to hide the markers at the thermometer control: • False entry hides the markers at the control. • True entry displays the markers at the control.
GoodRange	If the value falls into this range, the box gets the color of ’GoodRangeColor’. This entry allows a configuration of a range, e.g. *[20;100]*, or a complete range, e.g. *>=20*
WarningRange	If the value doesn’t fall into GoodRange, but into WarningRange, the box gets the color of ’WarningRangeColor’. his entry allows a configuration of a range, e.g. *[20;100]*, or a complete range, e.g. *>=20*
ErrorRange	If the values not within GoodRange or WarningRange, but within ErrorRange, the box gets the color of ’ErrorRangeColor’. This entry allows a configuration of a range, e.g. *[20;100]*, or a complete range, e.g. *>=20*

Table 10. Thermometer Control (Continued)

Field	Description
Color	Defines the color of the thermometer fluid, if *GoodRange* or *WarningRange* or *ErrorRange* is read from the process.
GoodRangeColor	Defines the color of the thermometer fluid for the range of values configured in *GoodRange*.
WarningRangeColor	Defines the color of the thermometer fluid for the range of values configured in *WarningRange*.
ErrorRangeColor	Defines the color of the thermometer fluid for the range of values configured in *ErrorRange*.
Dimension Left, Top, Width	The property entries' Left, Top, Width' can be used to adjust the label frame to a certain size.

Button Control

Button Controls allow to execute specific commands defined via DPV1 parameter manually by the user. Click on a button included in the DTM application to start a read or write access for the configured value.



Identifier	BT1
Visible	True
Enabled	True
Caption	Button BT1
Parameter	
Value	(Undefined)
Left	840
Top	1800
Width	1440
Height	480

Figure 45. Button Control

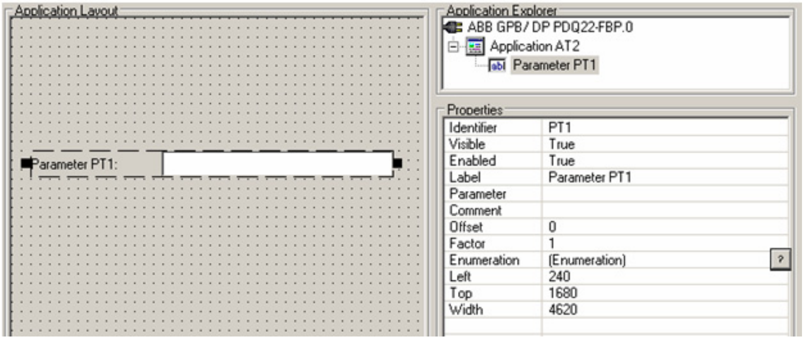
Properties:

Table 11. Button Control

Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Enabled	The control activates when the value is *true* and disabled when the value is *false*. By default, the value is *true*.
Caption	Include the label for the button, displayed in the application.
Parameter	The parameter defined in the DPV1 Tab and which will be executed by clicking the button. The parameter value will be updated with the device data or vice versa. A pull down menu shows all configured DPV1 parameter defined in the DPV1 Tab of the Configuration application.
Value	The value to write or read. The value must have the correct syntax of the required data type.
Dimension Left, Top, Width, Height	The property entries' Left, Top, Width, Height' can be used to adjust the label frame to a certain size.

Parameter Control

The Parameter Control can be used as standard parameter edit box, displays input and output device data defined in the DPV1 configuration. The appearance depends on the value type. It can be used as combo box or as list box.



Identifier	PT1
Visible	True
Enabled	True
Label	Parameter PT1
Parameter	
Comment	
Offset	0
Factor	1
Enumeration	(Enumeration)
Left	240
Top	1680
Width	4620

Figure 46. Parameter Control

Properties:

Table 12. Parameter Control

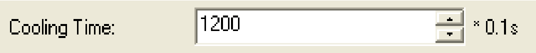
Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Enabled	The control activates when the value is *true* and disabled when the value is *false*. By default, the value is *true*.
Label	Name of the control. If empty, the gap left to the edit box disappears.
Parameter	The parameter, which shall be displayed or edited. A pull down menu shows all configured DPV1 parameter defined in the DPV1 Tab of the Configuration application.

Table 12. Parameter Control (Continued)

Field	Description
Comment	Text (e.g. engineering unit as constant) that appears right to the edit box. If empty, the gap right to the edit box disappears.
Offset	The Offset value is added to the raw value defined in the parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.
Factor	The value included in the factor property is multiplied with the raw value defined in the parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.
Enumeration	Opens the enumeration edit dialog. Here special values for this parameter can be defined. In User Interface these will appear in a drop-down box, where only the defined values can be selected. The parameter must be of a (unsigned) integer type.
Dimension Left, Top, Width	The property entries' Left, Top, Width' can be used to adjust the label frame to a certain size.

If the parameter is configured as data type boolean, a check box replaces the edit box. Numeric values additionally have an Up\Down button field. For strings and OctetStrings, a standard edit box is used. If the parameter cannot be found or it has an unknown type, the text box remains disabled. If the parameter is read-only, the edit box is also read-only.

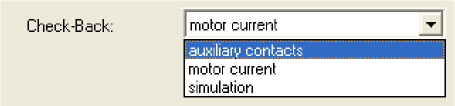
Example:



Cooling Time: 1200 * 0.1s

Figure 47. List Box

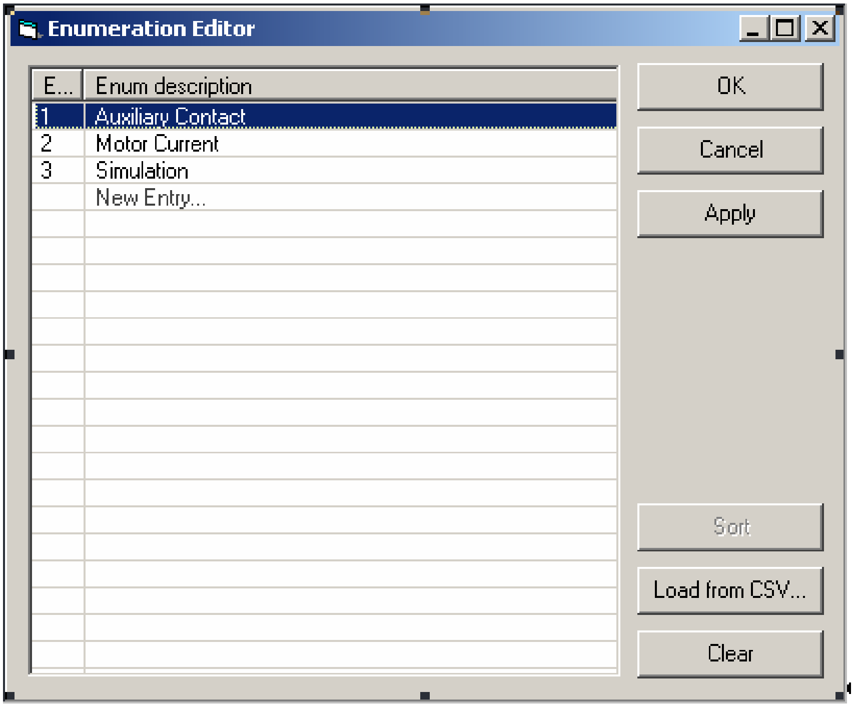
Parameter may have enumerated values as defined by the user. This is a useful functionality if a text, for example, Mode Automatic should be presented to the user



Check-Back: motor current

Figure 48. Combo box with Pull Down Menu

as a string “Automatic” instead of the value 1. In this case, the user has to configure the enumeration values in the Enumeration grid, which can be called through Enumeration property.



E...	Enum description
1	Auxiliary Contact
2	Motor Current
3	Simulation
	New Entry...

Figure 49. Enumeration Editor window

Enumerations can be included directly in the grid. They are loaded from a predefined CSV file. The CSV file must follow a specific syntax, otherwise it cannot be read in the grid. All defined enumerations in the CSV file are imported in the application file of the DTM.

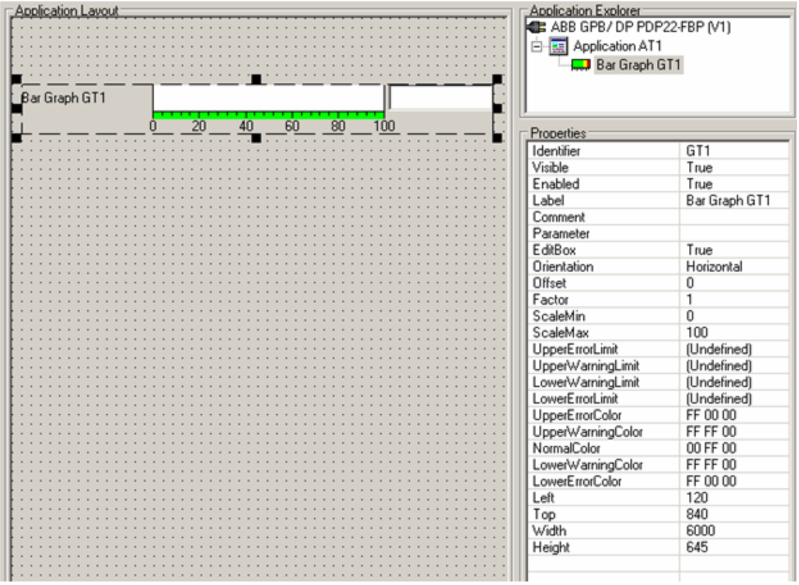
The header is mandatory for the DTM to identify a particular language.

Example of a “.CSV” file:

Value,	0x0409 (English),	0x0407 (German)	<Required header. To extend for next language insert language code 0x04xx (name)>
0	<definition>	<definition>	
1	<definition>	<definition>	
2	<definition>	<definition>	
...	...	...	

Bar Graph Control

Bar Graph Control allows to visualize measured value in a graphical way. It is for numeric data types only. Additionally, to the graphical display the measured value is shown as real value next to the bar graph.



Identifier	GT1
Visible	True
Enabled	True
Label	Bar Graph GT1
Comment	
Parameter	
EditBox	True
Orientation	Horizontal
Offset	0
Factor	1
ScaleMin	0
ScaleMax	100
UpperErrorLimit	(Undefined)
UpperWarningLimit	(Undefined)
LowerWarningLimit	(Undefined)
LowerErrorLimit	(Undefined)
UpperErrorColor	FF 00 00
UpperWarningColor	FF FF 00
NormalColor	00 FF 00
LowerWarningColor	FF FF 00
LowerErrorColor	FF 00 00
Left	120
Top	840
Width	6000
Height	645

Figure 50. Bar Graph Control

Properties:

Table 13. Bar Graph Control

Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Enabled	The control activates when the value is *true* and disabled when the value is *false*. By default, the value is *true*.

Table 13. Bar Graph Control (Continued)

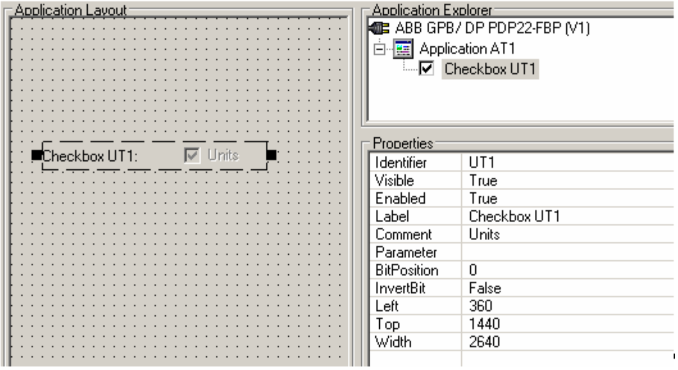
Field	Description
Label	Label of the measured value, which is visible left of the meter bar.
Comment	Text right of the edit box. Normally used for the unit.
Parameter	Parameter value, which shall be visible. A pull down menu shows all configured DPV1 parameter defined in the DPV1 Tab of the Configuration application. Value on the meter bar = Offset + Factor * Parameter Value.
Edit Box	The Edit Box (Value) field will be visible when the value is *true* and invisible when the value is *false*. By default, the value is *true*.
Orientation	Orientation will display the control in Horizontal or Vertical.
Offset	The Offset value is added to the raw value defined in the Parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.
Factor	The value included in the Factor property is multiplied with the raw value defined in the Parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.
ScaleMin	Lowest value on the meter bar ruler.
ScaleMax	Highest value on the meter bar ruler.
UpperErrorLimit	Value above UpperWarningLimit appears in the color of UpperErrorColor.
UpperWarningLimit	Values above UpperWarningLimit and below UpperErrorLimit appear in the color of UpperWarningColor.
LowerWarningLimit	Values below LowerWarningLimit and above LowerErrorLimit appear in the color of LowerWarningColor.
LowerErrorLimit	Value below LowerErrorLimit appear in the color of LowerErrorColor.

Table 13. Bar Graph Control (Continued)

Field	Description
UpperErrorColor	Value above UpperErrorLimit appear in the color of UpperErrorColor.
UpperWarningColor	Value above UpperWarningLimit and below UpperErrorLimit appear in the color of UpperWarningColor.
NormalColor	Value below UpperWarningLimit and above LowerWarningLimit appear in the color of NormalColor.
LowerWarningColor	Values below LowerWarningLimit and above LowerErrorLimit appear in the color of LowerWarningColor.
LowerErrorColor	Value below LowerErrorLimit appear in the color of LowerErrorColor.
Dimension Left, Top, Width	The property entries' Left, Top, Width' can be used to adjust the label frame to a certain size.

Check Box Control

The Check Box Control allows to sent to or read from the device a boolean value.



Identifier	UT1
Visible	True
Enabled	True
Label	Checkbox UT1
Comment	Units
Parameter	
BitPosition	0
InvertBit	False
Left	360
Top	1440
Width	2640

Figure 51. Check Box Control

The following data types are supported:

- integer.
- unsigned.
- boolean.
- OctetString.

Properties:

Table 14. Check Box Control

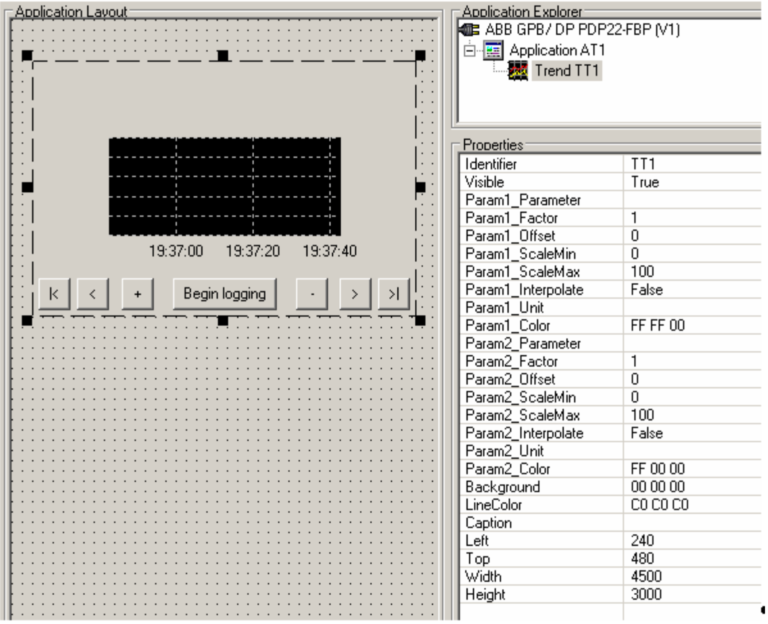
Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Enabled	The control activates when the value is true and disabled when the value is false. By default, the value is true.
Label	Label of the parameter value, which is visible left of the check box.

Table 14. Check Box Control (Continued)

Field	Description
Comment	Text right of the edit box. Normally used for the unit.
Parameter	The parameter to be used for the check box. A pull down menu shows all configured DPV1 parameter defined in the DPV1 Tab of the Configuration application.
BitPosition	Bit position to edit. ’0’ means the least significant bit. Use ’0’ for boolean type.
InvertBit	False entry handles the value *1* as checked. True entry handles the value *1* as unchecked.
Dimension Left, Top, Width	The property entries’ Left, Top, Width’ can be used to adjust the label frame to a certain size.

Trend Control

Trends allows to visualize up to two values simultaneously over a certain point of time. It is for numeric data types only. Additionally, the values can be saved in a log file.



Identifier	TT1
Visible	True
Param1_Parameter	
Param1_Factor	1
Param1_Offset	0
Param1_ScaleMin	0
Param1_ScaleMax	100
Param1_Interpolate	False
Param1_Unit	
Param1_Color	FF FF 00
Param2_Parameter	
Param2_Factor	1
Param2_Offset	0
Param2_ScaleMin	0
Param2_ScaleMax	100
Param2_Interpolate	False
Param2_Unit	
Param2_Color	FF 00 00
Background	00 00 00
LineColor	C0 C0 C0
Caption	
Left	240
Top	480
Width	4500
Height	3000

Figure 52. Trend Control

The Trend Control displays the history of one or two values. Param1 is the scaler on the left. Param2 is the scaler on the right.

Button Control in Trend Control:

Properties:

(<number> is the placeholder for parameter value 1 or 2)

Table 15. Button Controls

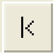
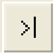
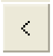
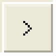
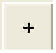
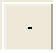
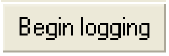
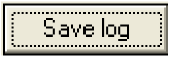
Button	Description
	Click on this button to jump to the first value in the trend display.
	Click on this button to jump to the last value in the trend display.
	Click on this button to scroll the trend view to the left.
	Click on this button to scroll the trend view to the right.
	Click this button to zoom in the trend display to get more details of a value.
	Click this button to zoom out the trend display to get a general view of the value (long time values).
	Click this button to start the log file record of the values, which are visible in the trend. After clicking this button the path and the file name in which the values shall be stored must be included.
	Click this button to save the recorded values to a CSV file on the hard disc.

Table 16. Trend Control

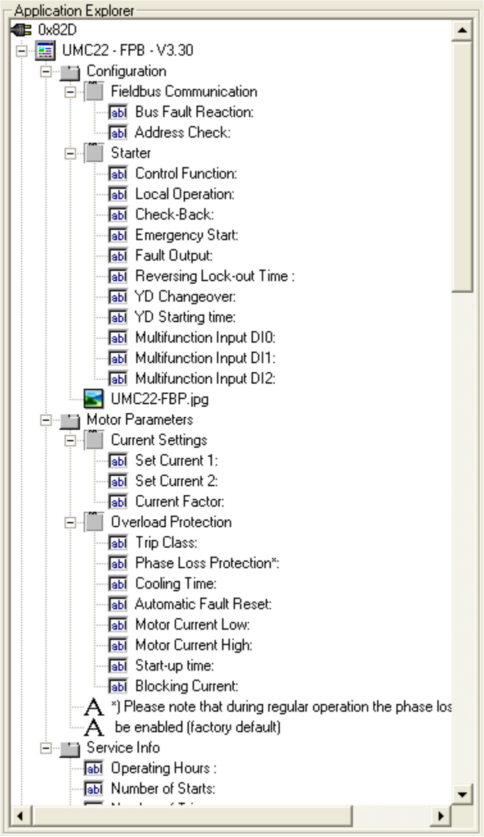
Field	Description
Identifier	The name of the application that is used to identify the application.
Visible	The Visible is used to display or hide the control.
Param<number>_Parameter	Parameter to poll. A pull down menu shows all configured DPV1 parameter defined in the DPV1 Tab of the Configuration application.
Param<number>_Offset	The Offset value is added to the raw value defined in the Parameter property. The offset is fixed to the configured parameter. The default value is 0, means no offset.
Param<number>_Factor	The value included in the Factor property is multiplied with the raw value defined in the parameter property. The factor is fixed to the configured parameter. The default value is 1, means raw value * 1.
Param<number>_ScaleMin	Lowest value on the scale.
Param<number>_ScaleMax	Highest value on the scale.
Param<number>_Interpolate	This property allows to interpolate trend values between two measured values. -> True value enables interpolation -> False value disables interpolation (step curves)
Param<number>_Unit	Unit of the transformed value. (e.g. “ml”, “kg”, “mol”
Param<number>_Color	This property defines the color, in which the trend line for this parameter will be visible.
Background	Color of the chart background
LineColor	Defines the vertical and horizontal line colors of the trend grid

Table 16. Trend Control (Continued)

Field	Description
Caption	Label for the trend, displayed as header for the trend in the DTM application.
<u>Dimension</u> Left, Top, Width	The property entries’ Left, Top, Width’ can be used to adjust the label frame to a certain size.

Application Explorer Area

The Application Explorer Area shows all configured and placed control in a structured form in the topology tree.



Application Explorer

- 0x82D
  - UMC22 - FPB - V3.30
    - Configuration
      - Fieldbus Communication
        - Bus Fault Reaction:
        - Address Check:
      - Starter
        - Control Function:
        - Local Operation:
        - Check-Back:
        - Emergency Start:
        - Fault Output:
        - Reversing Lock-out Time :
        - YD Changeover:
        - YD Starting time:
        - Multifunction Input DI0:
        - Multifunction Input DI1:
        - Multifunction Input DI2:
      - UMC22-FBP.jpg
    - Motor Parameters
      - Current Settings
        - Set Current 1:
        - Set Current 2:
        - Current Factor:
      - Overload Protection
        - Trip Class:
        - Phase Loss Protection*:
        - Cooling Time:
        - Automatic Fault Reset:
        - Motor Current Low:
        - Motor Current High:
        - Start-up time:
        - Blocking Current:
      - *) Please note that during regular operation the phase los
      - be enabled (factory default)
    - Service Info
      - Operating Hours :
      - Number of Starts:

Figure 53. Application Explorer Area with Example

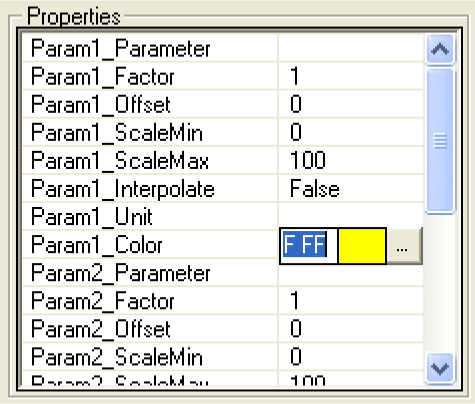
The icon is the root of the Explorer Area is labeled with device specific PNOID, for which the application is build. It has only symbolic functionality.

0x82D

Property Area

The Property Area allows to view and change the particular properties of the currently selected control. The listed properties depends on the type of selected control. The properties themselves are discussed in the Toolbox section. Color properties have a preview field when selecting the property. Other properties (including color properties) have a button at the right side, which open a file or color selection dialog. Instead of an edit box there can also be a drop down box for choosing predefined items.

Example:



Properties

Param1_Parameter	
Param1_Factor	1
Param1_Offset	0
Param1_ScaleMin	0
Param1_ScaleMax	100
Param1_Interpolate	False
Param1_Unit	
Param1_Color	F FF 🟨 ...
Param2_Parameter	
Param2_Factor	1
Param2_Offset	0
Param2_ScaleMin	0
Param2_ScaleMax	100

Application Script Editor Area

The Application Script Editor Area is the main workplace to create the logics that are used to manage the controls in the application.

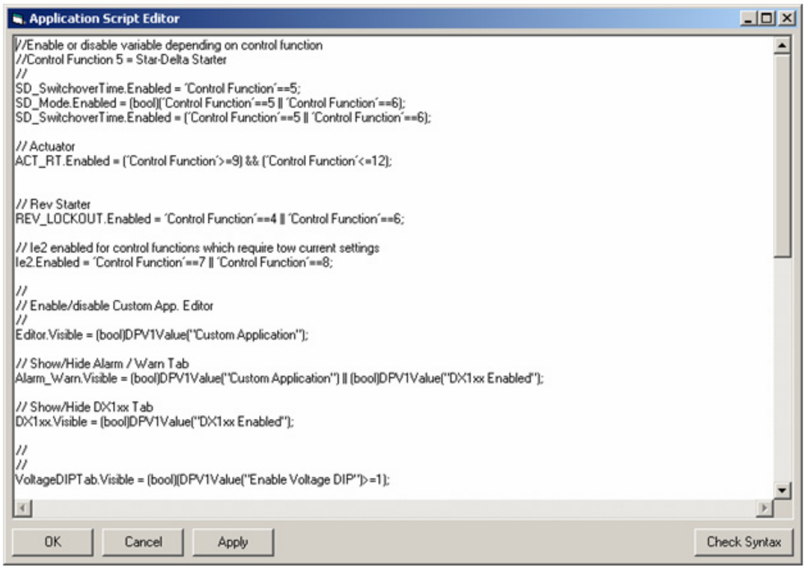
The operators used to create a logic are:

Table 17. Operators

Operator	Description
?:	The conditional operator takes three operands. It tests the result of the first operand and then evaluates one of the other two operands based on the result of the first.
!	The logical negation operator reverses the meaning of its operand
~	The one's complement operator, sometimes called the "bitwise complement" operator, yields a bitwise one's complement of its operand. The operand to the one's complement operator must be an integral type.
+	The ADD operator adds two numbers.
-	The SUB operator subtracts two numbers.
*	The MUL operator multiplies two numbers.
/	The DIV operator divides two numbers.
==	The Equality Operator tests the equivalence of two values.
!=	The Not Equality Operator tests the negated equivalence of two values.
>	The greater than Operator tests if the value of the left value is greater than that of the right.
<	The Less than Operator tests if the value of the left expression is less than that of the right.
>=	The greater than or equal to tests if the value of the left expression is greater than or equal to that of the right

Table 17. Operators (Continued)

Operator	Description
<=	The less than or equal to tests if the value of the left expression is less than or equal to that of the right.
<<	The << operator shifts the bits of expression1 left by the number of bits specified in expression2
>>	The >> operator shifts the bits of expression1 right by the number of bits specified in expression2
||	The logical OR operator indicates whether either operand is true.
&&	The logical AND operator indicates whether both operands are true.
|	The bitwise inclusive OR operator compares the values (in binary format) of each operand and yields a value whose bit pattern shows which bits in either of the operands has the value 1. If both of the bits are 0, the result of that bit is 0; otherwise, the result is 1. The values must be of type "Hex".
&	Bitwise AND operator compares each bit of its first operand to the corresponding bit of the second operand. If both bits are 1's, the corresponding bit of the result is set to 1. Otherwise, it sets the corresponding result bit to 0. The values must be of type "Hex".
^	The bitwise exclusive OR operator compares each bit of its first operand to the corresponding bit of the second operand. If both bits are 1's or both bits are 0's, the corresponding bit of the result is set to 0. Otherwise, it sets the corresponding result bit to 1. The values must be of type "Hex".



```
//Enable or disable variable depending on control function
//Control Function 5 = Star-Delta Starter
//
SD_SwitchoverTime.Enabled = 'Control Function'==5;
SD_Mode.Enabled = (bool)('Control Function'==5 || 'Control Function'==6);
SD_SwitchoverTime.Enabled = ('Control Function'==5 || 'Control Function'==6);

// Actuator
ACT_RT.Enabled = ('Control Function'>=9) && ('Control Function'<=12);


// Rev Starter
REV_LOCKOUT.Enabled = 'Control Function'==4 || 'Control Function'==6;

// Ie2 enabled for control functions which require tow current settings
Ie2.Enabled = 'Control Function'==7 || 'Control Function'==8;

//
// Enable/disable Custom App. Editor
//
Editor.Visible = (bool)DPV1Value("Custom Application");

// Show/Hide Alarm / Warn Tab
Alarm_Warn.Visible = (bool)DPV1Value("Custom Application") || (bool)DPV1Value("DX1xx Enabled");

// Show/Hide DX1xx Tab
DX1xx.Visible = (bool)DPV1Value("DX1xx Enabled");

//
//
VoltageDIPTab.Visible = (bool)(DPV1Value("Enable Voltage DIP")>=1);
```

Figure 54. Application Script Editor Area

The buttons available on the Application Script Editor are:

Table 18. Buttons in Script Editor

Button Name	Description
Check Syntax	It checks for the correctness of the syntax added. This includes checking for ControlIdentifier, Property, Parameter and so on.
OK	The Script data will save the changes and close the editor
Cancel	The Script Data will cancel the changes and close the editor
Apply	The data is saved into the editor.

Example

Procedure to create a sample logic in the Application Script Editor for managing a set of controls is given below:

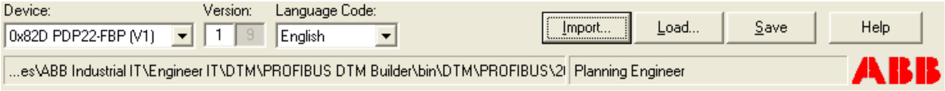
1. Create the new application in the PROFIBUS Application Editor.
2. Go to the Properties of the Application control. Then, set a caption, for example, Caption= “Testapplication", and assign the module.
3. Drag and drop a picture on to the application frame.
4. Go to the Properties of the picture and edit the Identifier and Visible. For example, Identifier= “TestPicture” and Visible= “False".
5. Drag and drop the Parameter control on to the application frame.
6. Go to the Properties of the Parameter control and edit the Identifier and Parameter. For example, Identifier = “TestParam" and Parameter= "PParam". Parameter should be of type "Byte".
7. Drag and drop the Check Box control on to the application frame.
8. Go to the Properties of the Check Box control and edit the Identifier, Enabled and Parameter. For example, Identifier = “TestCheck", "Enabled = False" and Parameter= "CCheck". Parameter should be of type "Bool".
9. Go to the Test application and in the Properties, click the ? next to Script.
10. The Application Script Editor gets opened.
11. Create the script in the editor. For example,

```
  TestPicture.Visible = `PParam`==Parametervalue
||(bool)DPV1Value("CParam");
```

12. Click [**Check Syntax**] to verify the syntax. A popup window with the message " No Syntax error found" appears if the syntax is correct.
13. Click [**Apply**]. Click [**OK**]. The data gets saved.
14. Click [**Save]** to save the created application.

To check the functioning of the logic, open the application created and use the controls as defined in the logic.

Button Control Area



Device: 0x82D PDP22-FBP (V1)

Version: 1 9

Language Code: English

...es\ABB Industrial IT\Engineer IT\DTM\PROFIBUS DTM Builder\bin\DTM\PROFIBUS\2

Planning Engineer

Figure 55. Button Control Area

The Button Control Area includes the following functions:

Table 19. Functions in Button Control Area

Field	Description
Device Area	This area lists all device types according GSD file, which have been imported in the Administration application by the user. The name of the device is shown as: <PNO_Ident_Num><Device Name in GSD> It is mandatory to select first the device, for which the application shell be created. The application is saved to a file containing the PNOID of the selected device type. If the specific device DTM is selected in the FDT frame, the created application will be taken for configuration.
Version Area	Allows to create an application for a specific version of the device. A change in the application editor results in a new minor version, which is included in the XML file. The minor version can not be changed by the user. It helps to have a history of changed application designs. Changing the major version results in a new XML file on the hard disk.
Language Code Area	For each application it is possible to set a specific language code. The language code is associated to the windows language settings. If the language is chosen in the Windows operating system, the particular application is shown. This offers the possibility to create different language application for the same device type.

Table 19. Functions in Button Control Area (Continued)

Field	Description
Import Button	Click this button to import an existing application file to this application.
Load Button	Click on this button to open an existing application file. The application, which was open before loading, is overridden from the new application during close or save.
Save Button	Click this button to save the configured or changed application. Each changed application will increase the minor version index after pressing the save button. The path to the folder, where the application is saved, is included in the status line.
Help Button	Click this button to get help for the Application Editor.

Application Layout Area

The Application Layout Area is the main workplace to place the required controls.

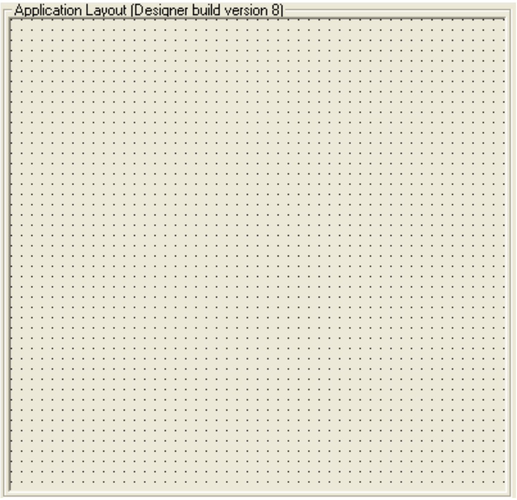


Figure 56. Application Layout Area

Data format and Storage

The newly created DTM application is stored in a standard XML file.

The XML root is a device tag which contains application tags. Application can either have a control tag that contains tags of all movable controls or a set of tab strip tags which represents tab strips.

The file name corresponds to the following pattern:

<DeviceID>_0_<MajorVersion>_<Language>_app_gpb.xml

where <DeviceID> is the unique type number for which the application was designed,

<MajorVersion> was chosen in the Application Editor and <Language> is a number for the language that the application has been translated (can also be chosen in the Application Editor).

Languages Codes are:

9 - English

7 - German

The file is saved into the project related path, which is defined by the FDT frame tool.

Index

Contact us

www.abb.com/800xA
www.abb.com/controlsystems

Power and productivity
for a better world™